SUNDAY AI · SEMICONDUCTOR STUDY GROUP

AI Infrastructure and Semiconductors

A seven-session self-study guide for readers with a background in statistics, computing or AI. Instead of memorizing terms, you set up a few equations and plug in the numbers yourself.

Edited by **Bowon Kwon**, Presiding Judge, Daegu District Court

Contents · How to use + 7 sessions + 3 appendices
Quantitative exercises · 32, with full solutions
Interactive figures · 12 (HTML edition; the print edition shows a still image in the same place)
Glossary index · 114 terms
Suggested pace · 8 weeks, about 2 hours a week

Printing · Press Ctrl+P and save as PDF; all collapsible solutions open automatically.
Offline · This single file is self-contained. The interactive figures work without an internet connection; links open when you are online.

Edition 2.0.1-en · translated from the Korean edition 2.0.1 (October 22, 2026) · compiled with generative AI. Errors may remain, and corrections will continue.

Contents

How to Use This Book 7
- What this book covers 8
- What each session contains 8
- An eight-week schedule 9
- Update the same picture every week 9
- When you get stuck 10
- A note on the numbers (read this) 10
- Read alongside other material 11

Mapping the AI Semiconductor Industry 12
- Learning objectives 12
- 0. Self-study guide — for readers working alone 13
- 1. Getting the intuition — what "AI semiconductor" actually means 14
- 2. The value chain — who makes money where 15
- 3. The shift in demand — from training to inference 16
- 4. How to read market figures critically 17
- 5. Three axes of supply-chain risk 19
- 6. Common misconceptions 20
- 7. Discussion questions 21
- 8. Further study 22
- Self-study appendix A — number-audit exercises and solutions 22
- Self-study appendix B — leads for the discussion questions 24
- Self-study appendix C — key terms 25
- Self-study appendix D — self-assessment 26
- Before the next session 27
- Sources 27

AI Chips — Compute Semiconductors 30
- Learning objectives 30
- 0. Self-study guide — for readers working alone 31
- 1. Getting the intuition — computing is cheap, delivering data is expensive 32
- 2. Why the CPU will not do 33
- 3. What a GPU really is — what one line of matmul does in hardware 34
- 4. TPUs and systolic arrays — building reuse into the hardware 35
- 5. NPUs and ASICs — the economics of specialization 38
- 6. The roofline model — the shared tool of this book 39
- 7. Software lock-in — a moat outside the technology 43
- 8. Common misconceptions 45
- 9. Quantitative exercises 46
- 10. Discussion questions 47

11. Further study 47
Self-study appendix A — solutions to the quantitative exercises 48
Self-study appendix B — leads for the discussion questions 56
Self-study appendix C — key terms 56
Self-study appendix D — self-assessment 57
Before the next session 58
Sources 58

AI Memory — DRAM · HBM · HBF 61
Learning objectives 61
0. Self-study guide — for readers working alone 62
1. Getting the intuition — the memory hierarchy is a map of physical compromises 63
2. The physics of DRAM — why it is slow and why it does not get faster 64
3. Two ways to raise bandwidth, and the dead end 66
4. HBM structure in detail 67
5. Why HBM is expensive — yields multiply 69
6. HBF (high bandwidth flash) — capacity, another bottleneck 71
7. PIM / PNM — taking computation to the memory 74
8. Common misconceptions 74
9. Quantitative exercises 76
10. Discussion questions 76
11. Further study 77
Self-study appendix A — solutions to the quantitative exercises 77
Self-study appendix B — leads for the discussion questions 84
Self-study appendix C — key terms 84
Self-study appendix D — self-assessment 85
Before the next session 86
Sources 86

Fabless, Foundry and the Value Chain 90
Learning objectives 90
0. Self-study guide — for readers working alone 91
1. Getting the intuition — why design and manufacturing split apart 93
2. The players and their roles 94
3. The chip-design pipeline — mapped onto software development 95
4. The truth about "3nm" — how to read node names 97
5. The economics of yield — this book's second key tool 98
6. ASIC, FPGA or general-purpose processor — a quantitative choice 101
7. Common misconceptions 102
8. Quantitative exercises 103
9. Discussion questions 104
10. Further study 104

Self-study appendix A — solutions to the quantitative exercises 105
Self-study appendix B — leads for the discussion questions 113
Self-study appendix C — key terms 113
Self-study appendix D — self-assessment 114
Before the next session 115
Sources 115

Front-End Processing 119

Learning objectives 119
0. Self-study guide — for readers working alone 120
1. Getting the intuition — chipmaking is "printing and coating," not "sculpting" 122
2. The wafer — the starting point 123
3. How the transistor structure evolved — from planar to FinFET and GAA 125
4. Lithography — the part we go deepest into 126
5. Etching — the subtractive process 130
6. Deposition — the additive process 132
7. Ion implantation, annealing and CMP 133
8. Inspection and metrology — the direct point of contact for CS and AI people 134
9. Putting it all together — how one layer is made 135
10. Common misconceptions 136
11. Quantitative exercises 137
12. Discussion questions 137
13. Further study 138
Self-study appendix A — solutions to the quantitative exercises 139
Self-study appendix B — leads for the discussion questions 147
Self-study appendix C — key terms 148
Self-study appendix D — self-assessment 150
Before the next session 150
Sources 150

Back-End Processing and Advanced Packaging 155

Learning objectives 155
0. Self-study guide — for readers working alone 156
1. Getting the intuition — the purpose of the back end has changed 157
2. Why the center of gravity moved — three pressures 158
3. Dicing — even cutting has become hard 161
4. TSV — vertical wiring that runs through the silicon 161
5. Interposers — the board that connects the chips 163
6. Bonding — a generational change in how chips are joined 166
7. 2.5D and 3D — the exact distinction 169
8. Heterogeneous integration and the chiplet standard (UCIe) 174
9. Signal integrity — familiar ground if you have a communications background 176

10. Common misconceptions 176
11. Quantitative exercises 177
12. Discussion questions 178
13. Further study 179
Self-study appendix A — solutions to the quantitative exercises 180
Self-study appendix B — leads for the discussion questions 190
Self-study appendix C — key terms 190
Self-study appendix D — self-assessment 192
Before the next session 192
Sources 193

AI Data Centers 197
Learning objectives 197
0. Self-study guide — for readers working alone 198
1. Getting the intuition — when density changes, everything changes 200
2. The power chain — from the grid to the transistor 201
3. The physics of cooling — why air is not enough 202
4. Reading efficiency metrics critically 204
5. Networks — scale-up and scale-out 206
6. Macro figures and their uncertainty 207
7. Back to the memory market — the last link 209
8. Common misconceptions 210
9. Quantitative exercises 211
10. Discussion questions 212
11. Further study 212
Self-study appendix A — Solutions to the quantitative exercises 213
Self-study appendix B — Leads for the discussion questions 229
Self-study appendix C — Key terms 230
Self-study appendix D — Self-assessment 231
Closing — the seven sessions in one line 231
Sources 232

Appendix I — Formula Summary and Exercise Index 236
1. Compute and bandwidth (Session 2) 236
2. Yield and cost (Session 4) 237
3. Lithography (Session 5) 237
4. Packaging and heat (Session 6) 238
5. Data centers (Session 7) 238
6. Index of quantitative exercises (all 32) 239
7. The seven sessions in one line 240

Appendix II — Glossary Index 241
Appendix III — List of Interactive Figures 246

Credits 250
About this book 250
How this book was made 250
Edition history 251
Acknowledgments 252
When you use this book 252

INTRODUCTION

How to Use This Book

BEFORE YOU START

The editor compiled this book with a generative AI system (Claude). Each session was checked against primary sources several times, but errors may remain. Please confirm figures and quotations against the originals. The book will keep being revised; the edition and date you are reading are on the cover and in the colophon (edition 2.0.1-en, translated from Korean edition 2.0.1 of October 22, 2026).

To the extent the editor holds rights in it, this book is released under CC BY-NC 4.0 (Creative Commons Attribution-NonCommercial 4.0 International). You may adapt and share it for noncommercial purposes if you credit the author, name the license (with a link), and indicate any changes you made. You may not restrict others' use with legal terms or technological measures. This summary is for convenience; the license text governs. The code of the interactive figures in the screen edition (HTML, CSS and JavaScript) is released separately under the PolyForm Noncommercial License 1.0.0. Noncommercial use, modification and redistribution are permitted on that license's terms; for commercial use, please contact the editor in advance through the repository's issue tracker to agree on separate terms. Tables, quotations and figures reproduced or redrawn from original works are not covered by these licenses and remain subject to the rights of their authors. If you hold rights in such material, tell us through the repository's issue tracker and we will correct or remove it.

This English edition is a translation of Korean edition 2.0.1 (October 22, 2026). The editor made it with the same AI system and had it checked against the Korean text and, where a quotation was originally in English, against that original wording. Its numbers keep the Korean edition's reference date of October 8, 2026 (see "A note on the numbers" below). Sources that exist only in Korean are listed with translated titles and marked "(in Korean)."

This edition keeps the Korean edition's Korean examples as they are. In a few places where a Korean case carries the argument, a box headed "Comparable cases elsewhere" adds an example from another country. Those examples may not fit your jurisdiction, your date or your industry, so check the source cited in the box before you rely on one.

This is a self-study guide compiled for the semiconductor sessions of *Sunday AI*, a study group, and covers AI infrastructure and semiconductors in seven sessions. It is written for readers with a background in statistics, computing or AI who never studied semiconductors. Instead of asking you to memorize terms, it **sets up a handful of equations and has you plug in the numbers yourself.** Every session comes with calculation exercises and full solutions, so you can check your answers on your own. In the screen edition (HTML), interactive figures placed next to the equations let you change the values; the print edition (PDF) carries the same figures in their initial state at the same spots. You are welcome to read it the same way in another group, or alone.

What this book covers

The seven sessions follow one question: **"Where are the limits of the physical equipment that runs AI set?"**

SESSION	TOPIC	TOOL THE SESSION BUILDS
1	How the AI semiconductor industry is laid out, and how to read its numbers	The habit of asking for the definition first
2	Compute chips (GPU, NPU, TPU, ASIC)	**The roofline model**
3	Memory (DRAM, HBM, HBF)	Separating bandwidth from capacity
4	Fabless, foundries and the value chain	**The yield equation**
5	Front-end processes (lithography, etching, deposition, metrology)	**The Rayleigh criterion**
6	Back-end processes and advanced packaging	Chiplet trade-offs, the thermal-resistance chain
7	AI data centers (power and cooling)	Synthesis

The two equations set up in Sessions 2 and 4 govern everything else. If you are short of time, work through at least those two by hand.

What each session contains

Every session follows the same structure.

ORDER	NAME	WHAT IT DOES
1	Key takeaway	A three-line preview of the session's conclusion
2	Learning objectives	Questions you should be able to answer at the end
3	**0. Self-study guide**	**For readers working alone.** What to bring from the previous session, a prerequisite check, **reading order and time budget**, and detours around the usual sticking points
4	Main text	
5	Common misconceptions	Things that are often stated wrongly
6	Quantitative exercises	Problems to solve by hand
7	Discussion questions	Questions without a single right answer
8	**Self-study appendix A**	**Full solutions to the exercises.** They are folded; solve first, then open them

ORDER	NAME	WHAT IT DOES
9	Self-study appendix B	The axes along which the discussion questions divide
10	Self-study appendix C	Key terms of the session
11	**Self-study appendix D**	**Self-assessment checklist.** Do not move on to the next session until you pass the ★ items
12	Before the next session	Open questions paired with where the next session answers them

The solutions in appendix A are folded. On screen, click the heading to open them; when you print, they all open automatically.

An eight-week schedule

There are seven sessions, but **eight weeks works better.** Session 5 (front-end processes) is one and a half times as long as the others.

WEEK	READ	TIME	MUST SOLVE BY HAND THAT WEEK
1	Session 1, how the AI semiconductor industry is laid out	100 min	The three number-audit problems
2	Session 2, AI chips	120 min	**Exercises 1 and 2**
3	Session 3, AI memory	110 min	**Exercises 1 and 2**
4	Session 4, fabless and foundries	110 min	**Exercises 1, 2 and 4**
5	Session 5, front-end processes, first half (sections 1–4)	95 min	**Exercises 1 and 2**
6	Session 5, front-end processes, second half (sections 5–9)	65 min	**Exercises 3, 4 and 5**
7	Session 6, back-end processes and advanced packaging	130 min	**Exercises 1, 2 and 4**
8	Session 7, AI data centers, plus synthesis	190 min	**Exercise 5** (required)

That is about two hours a week. If one session is too much at a sitting, break it at the steps in that session's "Reading order" table.

Update the same picture every week

What breaks down most easily in self-study is **"I read it, but nothing stayed."** There is one remedy. Draw the skeleton below on one sheet of paper (or one screen), and **add the layer you learned each week on top of it.**

```
[request] → [model/software] → [compute chip] → [memory] → [package]
                                                                  ↓
[power/cooling] ← [DC/rack] ← [server/board] ← [fab/supply chain] ↵
```

Three rules for updating it

1. **Put one number on every arrow.** (For example, on the arrow from compute chip to memory: "ridge point ≈ 295 FLOP/byte".)
2. **Mark on every box whether it is the bottleneck.** As the sessions go on, you should see the bottleneck mark move.
3. **Do not erase last week's picture.** Draw over it in a different color.

"The seven sessions in one line" at the end of Session 7 is the final answer key. **If in week 8 you can reconstruct it without looking, you are done.**

When you get stuck

SYMPTOM	CAUSE	WHAT TO DO
Everything from Session 2 on feels like it slides past	You have not digested "the whole set of weights is read" in Session 1, section 3	Go back to the **library analogy** in Session 1's self-study guide. If this does not land, nothing after it will
I can follow the equations but not why they matter	You have not plugged in numbers	Do appendix A **with the solutions covered.** The book is designed so that reading without calculating leaves nothing behind
Too many terms	You are trying to memorize them all at once	Do not memorize; keep the **Appendix II glossary** beside you and look terms up as they come
The lithography part of Session 5 will not go in	Not enough optics background	Start with the **camera-aperture analogy** in Session 5's self-study guide. If that fails, read only sections 3 and 4, move on, and come back after Session 6
The volume is overwhelming	You are trying to read it all at once	Break it at the **steps** in each session's "Reading order" table
I read everything but cannot connect it	You read session by session without building the chain	Start the **cumulative picture** above, even now

---

A note on the numbers (read this)

The calculation exercises in this book use many **assumed values.** They are examples meant to show the direction of a conclusion, not measurements.

The reference date is October 8, 2026. Statements that change over time, such as volume-production and shipment status, power-demand projections, standards and product

specifications, were rechecked on that date against primary sources (manufacturer announcements, standards documents and agency reports), and the text gives the date each was checked.

Assumed values that must not be quoted as facts

- The price of one wafer (USD 15,000), electricity (USD 0.08 per kWh), the price of an accelerator node (USD 250,000)
- ASIC and GPU unit prices and NRE, defect density D_0, assembly yield, interface overhead ratios
- Any other settings stated in individual exercises

Values taken from public spec sheets that change with each generation

- Compute performance and memory bandwidth of accelerators (A100, H100, B200). All are **dense (no sparsity)** figures. Vendor material also lists a figure twice as high with sparsity, so always check which one you are looking at
- HBM specifications, the reticle limit, microbump and hybrid-bonding pitch

Things that are structurally stable and safe to quote

- The equations themselves: the roofline, the yield model, the Rayleigh criterion, thermal resistance, the definition of PUE
- **The direction of the relationships** that follow from them: yield falls exponentially as die area grows, depth of focus worsens with the square of NA, small batches are bandwidth-bound, and so on

This book deliberately leaves out absolute figures for market size and market share. Each source defines and counts them differently, so they vary widely. Section 4 of Session 1 explains why.

Read alongside other material

The thinner parts of this book are the following.

TOPIC	COVERAGE HERE
AI compilers and the software stack	Session 2, section 7 treats them only from the angle of lock-in. There is almost nothing on the compilation pipeline itself — **the largest gap**
Links to primary sources	Links are given in the "Primary sources" box at the head of each session and in each session's sources section. Paid standards and papers are given with bibliographic data and DOI only
Diagrams and images	There are redrawn diagrams and interactive figures, but no photographs or micrographs

SESSION 1

Mapping the AI Semiconductor Industry

KEY TAKEAWAY

"AI semiconductors" is not a single product but a loose category that spans compute chips and memory chips, and that vagueness is the first trap when you read market statistics. The industry splits into three layers: design (fabless), manufacturing (foundry/IDM), and equipment and materials. **The layer that earns the most and the layer that cannot be replaced are not the same.** Demand is moving from training to inference, and that shift has pushed the memory bottleneck to the center of the whole industry. The goal of this session is not to memorize numbers but to get into the habit of checking which definition a number stands on.

PRIMARY SOURCES

KDB Future Strategy Research Institute, "AI Semiconductor Technology and Industry Trends" (in Korean; KDI Economic Information and Education Center) · Samil PwC, "Global Semiconductor Industry Outlook 2026" (in Korean, PDF) · PwC Consulting Korea, "The Paradox of the Semiconductor Supercycle" (in Korean) · LBNL press release on its 2024 U.S. data center energy report · NVIDIA DGX H100/H200 User Guide — power specifications · eCFR 15 CFR §734.9 (FDP rules)

Learning objectives

By the end of this session you should be able to answer the following.

1. When you see a news story saying "the AI chip market is worth trillions of dollars," what should you check first?
2. How do profit margins and irreplaceability differ across the layers of the value chain?
3. As demand moves from training to inference, how do the hardware requirements change?
4. Into what types do semiconductor supply-chain risks fall, and how do they differ in character?

0. Self-study guide — for readers working alone

> **THIS IS A READING SESSION**
>
> It is the only one of the seven sessions with no calculations. Its real goal is not to acquire knowledge but **to build one habit**: whenever you see a number, first ask "what was counted to get this?" Section 4 is the training ground; the other sections set the stage on which you will use the habit.

Prerequisite check — can you explain each of these five in one line?

If any of them stops you, read the answer below and move on. That is all you need for this session.

CONCEPT	ONE-LINE ANSWER
Parameter	Each number the model fixed through training (a weight). A "70B model" has 70 billion of them
FLOP / FLOP/s	FLOP is a **count** of floating-point operations; FLOP/s is the count per second (**speed**). The first is an amount of work, the second a processing capacity
Memory bandwidth	The number of bytes per second that can be moved from memory to the chip [byte/s]. A completely different notion from capacity (GB)
Wafer	A disk of silicon. Hundreds of chips are made on it at once and cut apart at the end
Fabless / foundry	A company that only designs chips / a plant that manufactures other companies' designs under contract

Reading order (100 minutes recommended)

When you read alone, front to back is not the best order.

STEP	SECTION	TIME	WHY
1	Section 1, getting the intuition	15 min	Fixes the scope of the term first
2	**Section 3, from training to inference**	**25 min**	**The heart of this session.** The other six sessions all grow out of it
3	Section 2, the value chain	15 min	After section 3 you can see why money gathers in this layer
4	Section 4, reading the numbers	20 min	Together with the exercises below
5	Sections 5 and 6, risks and misconceptions	15 min	
6	Self-study exercises	10 min	

Never skip: "Why this shift put memory on the throne" in section 3. The single line "arithmetic intensity $I \approx B$" found there leads straight to the roofline in Session 2, HBM in Session 3, packaging in Session 6, and the power problem in Session 7. **If this paragraph**

does not make sense, Session 2 will keep sliding past you.

If section 3 is hard — a detour

Section 3 borrows ideas that have not yet been set up (arithmetic intensity, ridge point), so getting stuck on a first reading is normal. Get a feel for it with the analogy below, move on, and **come back** after reading Session 2. It will open up.

> Suppose you fetch an entire book from the library, read one sentence, and send it back. Moving the book takes ten minutes; reading takes one second. The bottleneck is not "reading speed" but "moving speed." Decoding during inference is exactly like this. To produce **one** token, the **whole** set of model weights is read from memory. However fast the chip is, it makes no difference; how fast you can haul the weights in from memory decides everything. There is only one fix: once the book has been brought over, **have many people share the reading** (batching). This is where the importance of batching comes from.

---

1. Getting the intuition — what "AI semiconductor" actually means

It is not one thing

The term "AI semiconductor" refers to at least four things.

WHAT PEOPLE CALL IT	THE ACTUAL OBJECT	TYPICAL EXAMPLES
Narrow sense (compute)	Accelerator chips specialized for AI computation	GPU, TPU, NPU, inference ASIC
Broad sense (compute + memory)	The above plus memory for AI	+ HBM
System sense	The whole accelerator module or board	HGX board, accelerator card
Industry sense	Every semiconductor sold because of AI demand	+ network chips, power semiconductors, CPUs

When an article or report says "the AI chip market is worth X billion dollars," it often does not say which of the four it means. **Different definitions produce numbers that differ several-fold.** The first habit this book tries to build is asking "which scope is this?" every time you see a number.

Why this industry suddenly grew so large — three lines

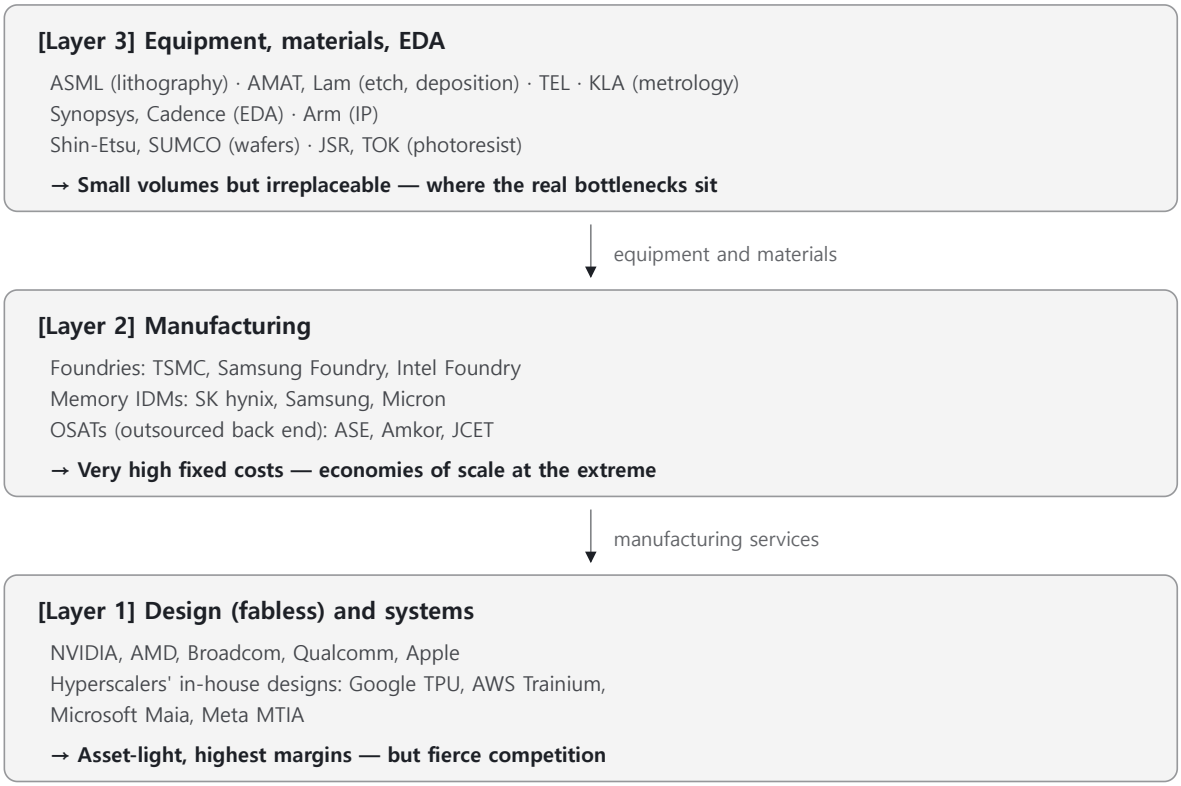
1. **A transformer is a lump of matrix multiplications.** Its computation reduces to massively parallel multiply-accumulate operations, so the gain from dedicated hardware is extreme.
2. **Scaling laws gave empirical support to "bigger model = better performance."** The belief that compute input translates into performance justified the capital spending.
3. **But only a handful of places can make that hardware.** Demand exploded while supply had no physical or financial elasticity. So prices and profits piled up at particular points.

You can read this book as a seven-session dissection of the "why only a handful" in point 3.

---

2. The value chain — who makes money where

Three layers



Between layers 1 and 2 of the figure sits one more thin layer. **Design houses** (design service companies) take the logical design handed over by a fabless company, implement and verify it physically for a particular foundry's process, and pass it on to manufacturing. Foundries tie them into partner networks. TSMC describes the members of its VCA (Value Chain Alliance; formerly Value Chain Aggregator) as "independent design service companies working closely with TSMC to help system companies, ASIC companies, and emerging start-up customers bring their innovation to production." They "provide specific services at each

link in the IC value chain, including IP development, design backend, wafer manufacturing, assembly, and testing" (checked October 2026). The players and each foundry's programs are covered in Session 4.

Key insight — profit margin and irreplaceability are different things

- **Highest margins:** the leading fabless companies in layer 1 and the leading foundry in layer 2
- **Highest irreplaceability:** layer 3. EUV lithography tools in particular come **effectively from ASML alone**. Its revenue is far smaller than NVIDIA's, but if it stops, all of leading-edge logic stops.

This asymmetry explains why geopolitical controls do not stop at finished chips but also aim at layer 3 (equipment). Controls on finished chips block the flow of chips sold today; controls on equipment tie up the ability to make such chips yourself (section 5 ③).

Where Korea stands

This book was written for a study group in Korea, so the original keeps a short note on the country's position. It is kept here because it shows how the three-layer picture reads from one country's vantage point.

- **Strong:** memory (DRAM, NAND, HBM). SK hynix and Samsung Electronics hold a large share of the world market. For example, by revenue in the first quarter of 2026, SK hynix ranked first in the world in HBM and second in DRAM (including HBM) and in NAND (IDC figures, as given in the prospectus SK hynix filed for its U.S. listing in July 2026; checked 2026-10-08).
- **Relatively weak:** share of leading-edge foundry, the fabless ecosystem, the domestic share of equipment and materials.
- **Structural implication:** memory has traditionally been a market with violent cycles. Whether AI demand has changed that cycle structurally or only widened its swings is the final discussion topic of Session 7.

> **Comparable cases elsewhere.** Taiwan reads the same picture from layer 2. TSMC reported that it "represented 34 percent of the Foundry 2.0 industry, which we define as all logic wafer manufacturing, packaging, testing, mask-making and others, output value in 2024, as compared to 28 percent in the previous year" (TSMC 2024 business report, in the minutes of its June 3, 2025 shareholders' meeting). The definition is the company's own, so the share is also a small case of the problem in section 4: change the denominator and the number changes. (source)

---

3. The shift in demand — from training to inference

The concern behind every later session comes from this section.

Training and inference want different machines

ITEM	TRAINING	INFERENCE
Computation pattern	Large-batch matrix multiplication (GEMM)	Decoding is close to matrix-vector multiplication
Arithmetic intensity	High (hundreds of FLOP/byte)	**Low** (on the order of the batch size)
Main bottleneck	Compute performance, communication between nodes	**Memory bandwidth, capacity, latency**
Precision	BF16/FP8	INT8/FP8/INT4 (aggressive quantization)
Memory needs	Weights + activations + optimizer state	Weights + **KV cache**
Latency sensitivity	Low (batch processing)	High (a user is waiting)
Cost structure	One-off large investment	**Ongoing operating cost**

Why this shift put memory on the throne

Training uses large batches, so it can keep the compute units busy. But **in the decoding phase of inference, producing one token requires reading the whole set of model weights from memory.** Each read of the weights supports only as many operations as there are sequences in the batch.

Roughly speaking, the amount of computation per byte of fp16 weights is proportional to the batch size B. Arithmetic intensity is about $I \approx B$. As Session 2 will show, the ridge point of current GPUs is around 300 FLOP/byte. **With a batch below 300, the GPU is bound by bandwidth and its compute units sit idle.**

Yet latency requirements keep the batch from growing without limit, and the KV cache eats memory, so batch size is capped by memory capacity. In the end, the economics of an inference service are decided **not by "how fast the chip is" but by "how wide and how large a memory is attached to it."**

> **This is why HBM became the center of the industry (Session 3), why advanced packaging became a bottleneck (Session 6), and why data center power became a problem (Session 7).**

---

4. How to read market figures critically

What actually happens

For the "AI accelerator market" of the same year, different sources give very different figures. The reasons:

1. **Differences in scope** — Is it the accelerator chip only, the module and board too, or the whole AI server? Counting the whole server brings in the CPU, memory, network and

power supply, and the number grows several-fold.

Power figures hide the same trap. NVIDIA lists the maximum thermal design power (TDP) of the H100 (SXM) as "Up to 700W (configurable)", and gives the maximum input power of the DGX H100 server, which holds eight H100s, as 10.2 kW. Do the arithmetic: the TDP of eight chips adds up to 5.6 kW, and the gap to the server's maximum input power is 4.6 kW. Items such as the CPUs, memory, network, fans and power-conversion losses fall into that gap. Both values are ceilings, however, so this does not mean that 4.6 kW is used outside the chips during actual work. This session does not break the gap down into numbers. The next-generation DGX B200 has a maximum system power of about 14.3 kW. Each time you widen the boundary, from chip to server and from server to a facility that includes cooling and power distribution, the number grows. When you see the phrase "the power one AI chip uses," first ask at which boundary it was measured. The question comes back in the power-cost calculation of Exercise 4 in Session 2. Manufacturing has the same structure: wafer capacity, number of good dies, package output and shipments are different numbers.

2. **Differences in the unit counted** — Revenue, units shipped, or deployed compute (FLOPS)? In-house chips (such as TPUs), for example, are not sold outside, so they are underestimated in revenue-based statistics and come out much larger when counted in deployed FLOPS.

3. **The interests of whoever publishes** — chip companies, market research firms and investment banks each have different incentives.

4. **Differences in timing** — this is a market whose forecasts are revised heavily every half year.

What can and cannot be pinned down

This book draws the line as follows.

Structurally settled (several sources agree)

- NVIDIA holds the overwhelming majority of data center AI accelerator revenue. What the ratio is and which way it is moving depends on how you set the denominator, so it is not given here (Exercise 1).
- The share of hyperscalers' in-house ASICs (Google TPU, AWS Trainium, Meta MTIA, Microsoft Maia) is rising fast, and Broadcom, Marvell and others have risen as their design partners.
- Inference has overtaken, or is overtaking, training as a share of AI compute spending.
- HBM is an oligopoly of three companies (SK hynix, Samsung, Micron), and SK hynix leads by revenue (confirmed with IDC figures for the first quarter of 2026, 2026-10-08). The order and gap between second and third place swing with each product generation, so they are not stated here.

Not settled (sources vary widely — mention only as ranges)

- Absolute market size by year
- Exact percentage shares by company (for example, the same company's HBM share differs by several percentage points depending on whether it is a quarterly result or an annual forecast, and whether it is counted by revenue or by shipments)
- Total data center power consumption. Even authoritative estimates span a wide range. Lawrence Berkeley National Laboratory (LBNL) estimated U.S. data center electricity use in 2028 at 325–580 TWh (section 6, misconception 6)

> **If you read this in a group:** make it a rule that the presenter must state "the source and definition of this number in one line." The learning effect of this session rises sharply.

---

5. Three axes of supply-chain risk

① Geographic concentration

Leading-edge logic manufacturing is heavily concentrated in Taiwan. That makes it a single point of failure, both for natural disasters and for geopolitics. Countries' policies to attract fabs (the U.S. CHIPS Act, the EU Chips Act, support in Japan and Korea) try to ease this concentration, but they are not short-term fixes: building one fab and raising its yield takes years.

② Single-supplier bottlenecks

BOTTLENECK	EFFECTIVE SUPPLIER
EUV lithography tools	ASML (effectively the only one)
Advanced packaging (CoWoS and similar)	Centered on TSMC
EDA tools	Synopsys, Cadence, Siemens EDA
CPU IP	Arm (licensing model)
High-purity wafers and photoresist	Concentrated in Japanese firms

The important point here is that **"can it be made technically" and "how much can actually be made" are different questions.** For instance, the supply of AI accelerators runs into limits on CoWoS packaging capacity and HBM supply before it runs into limits on logic wafers. (Covered in detail in Session 6.)

Bottlenecks also form a chain on the demand side. EUV → CoWoS → HBM is the movement of the bottleneck seen from the supply side. The same movement can be traced from the demand side. As models grow and contexts lengthen, both compute and the KV cache grow, and so do the capacity and bandwidth demanded of the HBM that holds the weights and the KV cache. Attaching more HBM makes the package larger and more complex. Pack such accelerators densely into servers and the heat density rises, until the power and cooling of

the racks and the facility reach their limits. When demand at one link rises, the pressure passes to the next. So what constrains the system is usually not performance within one layer but **the interfaces where layers meet**: between memory and chip, chip and package, server and facility. Session 3 (HBM), Session 6 (packaging) and Session 7 (servers and facilities) each take one of these links.

③ **Policy risk**

Export controls target **equipment and manufacturing capability** as well as finished chips. Because of extraterritorial reach (the foreign direct product rules and others), firms in third countries are affected too. As of October 2026, the text of the foreign direct product (FDP) rules in the U.S. Export Administration Regulations (EAR) (15 CFR §734.9) contains a separate advanced computing FDP rule (paragraph (h); advanced integrated circuits and computers) and a semiconductor manufacturing equipment (SME) FDP rule (paragraph (k)). The structure is that an item made with certain U.S. technology or software, or made in a plant built with such technology, can be subject to the EAR even if produced outside the United States. The text can be read through the eCFR link under "Further study" in section 8.

"The paradox of the supercycle"

The stronger the boom, the more everyone invests in expanding the bottleneck process. Because fabs and equipment have lead times of two to three years, the demand signal at the moment of the investment decision and the demand at the moment the capacity comes online diverge. This is the semiconductor industry's classic bullwhip effect, and it is especially severe in memory. **One camp argues that AI demand has abolished this cycle; the other, that it has only widened the swings.** (Final discussion of Session 7.)

---

6. Common misconceptions

Misconception 1: "AI chip = GPU" → GPUs dominate today, but the share of ASICs such as TPU and Trainium is rising fast. More important, memory (HBM) and packaging account for a large part of the performance and cost of an AI system.

Misconception 2: "The smaller the nanometer number, the better the chip, always" → Node names are marketing labels, not physical dimensions (Session 4). AI accelerator performance also depends more on memory bandwidth and packaging than on the node.

Misconception 3: "With enough money, you just build a fab" → Equipment lead times, skilled labor and the yield ramp (the learning curve to reach volume-production yield) are bigger constraints than capital. A fab building goes up in two years, but bringing yield to a competitive level takes longer.

Misconception 4: "China will soon catch up / will never catch up" → Both are oversimplifications. China has already secured considerable capacity at mature (legacy) nodes, while at the leading edge of logic it faces the structural constraint of having no EUV. It has to be looked at layer by layer.

Misconception 5: "The AI chip boom will end soon / will last forever" → The honest answer is: "Because the nature of demand has shifted from training to inference, it is uncertain whether the old cycle models still apply." Inference demand is more continuous and more like an operating expense than training, which could dampen the cycle, but the structure of capacity lead times is unchanged.

Misconception 6: "Once power-efficient chips arrive, total data center power will fall" → The efficiency of one chip and the total for a whole facility are different questions. According to LBNL's 2024 report on U.S. data center energy use, prepared with support from the U.S. Department of Energy, U.S. data center electricity use grew from 58 TWh in 2014 to 176 TWh in 2023, about 4.4% of total U.S. electricity in 2023. For 2028 it is estimated at 325–580 TWh, 6.7–12% of U.S. electricity (the share depends on how fast the rest of the economy grows). Use more than doubled between 2017 and 2023 alone, and the report names the growth of AI servers as the main cause. Over the same period, chips became more power-efficient with each generation. Yet the total tripled, so efficiency and total can move separately (this interpretation is this book's own, not a conclusion taken from the report). Note also that the 2028 estimate spans nearly a factor of two. That is why section 4 classes the total as "not settled" and gives it only as a range.

---

7. Discussion questions

1. From a country's point of view, in which of the three layers of the value chain is it most efficient to put investment and policy resources? How does the answer change with the time horizon (5 years vs. 20 years)?
2. Is it rational for hyperscalers to build their own chips? Under what conditions is it rational, and under what conditions not? (Taken up quantitatively in Session 4.)
3. Are the wide differences among estimates of "the AI chip market size" a problem of measurement or a problem of definition? As someone trained in statistics, how should you treat these estimates?
4. How should we understand the bottleneck in the supply chain moving from EUV (equipment) to CoWoS (packaging) and then to HBM (memory)? Why does the bottleneck keep moving?
5. Is there evidence that the traditional memory cycle (boom → expansion → glut → slump) has been changed structurally by AI demand?

---

8. Further study

- **Overview of the industry structure:** Chris Miller, *Chip War* (2022) — the standard popular book on the geopolitics of the semiconductor industry. Good background reading for this session
- **Regular tracking of technology trends:** industry analysis from SemiAnalysis, TrendForce, Counterpoint Research and others. Allow for each one's slant
- **Primary material:** quarterly earnings materials from TSMC, SK hynix, Micron and NVIDIA (especially the Q&A of the earnings calls). Often more accurate than market reports
- **Korean perspectives (in Korean):** semiconductor industry analysis from the KDI Economic Information and Education Center, the Korea Institute for Industrial Economics and Trade (KIET), and ETRI's *Electronics and Telecommunications Trends*
- **Power demand:** LBNL press release on the 2024 U.S. data center energy report — source of the figures in section 6, misconception 6
- **Export-control text:** eCFR 15 CFR §734.9 — current text of the foreign direct product (FDP) rules

---

Self-study appendix A — number-audit exercises and solutions

This session has no calculation exercises. Instead it has three problems that put the "ask for the definition first" habit from section 4 to work. **Write your answer on paper before you look.**

Exercise 1 — Same company, different numbers

One article says "Company A's share of the AI accelerator market is 90%," and another says "70%." Assuming neither article is lying, what differences in definition could produce this gap? Give at least three.

▾ Solution

There are three or more legitimate differences in definition.

1. **Whether in-house chips are in the denominator.** Google TPU and AWS Trainium are not sold outside, so they barely register in a **revenue-based market.** Take in-house chips out of the denominator and Company A's share goes up; put them in on a deployed-units or deployed-compute basis and it goes down. This alone produces a 20-percentage-point difference.
2. **Chips only, or systems too.** Shares at the accelerator-die level and at the server or rack level differ. A whole server includes the CPU, network and power supply, and

the mix of suppliers for those items is different.

3. **Training only, or inference too.** The inference market is fragmented, with CPU inference, low-cost accelerators and edge NPUs mixed in. Looking at the training market alone gives a much higher concentration.
4. (Extra) **Timing.** Quarterly shipments or cumulative installed base? In a fast-growing market the two diverge widely.

What to take away: a share is decided not by the numerator but **by the denominator.** If an article does not state the denominator, the number is an unverifiable statement.

Exercise 2 — The growth-rate trap

In the same year, one article writes "the AI chip market grew 200% year on year," and another writes "the semiconductor market grew 25% year on year." Both are about the semiconductor industry. Is this a contradiction? Before answering, write down what each sentence counted.

▾ Solution

It is not a contradiction. It **mixes a part with the whole.**

Check it with a simple numerical example. Say that out of a total semiconductor market of 100, AI-related sales are 5 and the rest 95. If AI triples to 15 (+200%) and the rest grows about 16% from 95 to 110, the total goes from 100 to 125, **a 25% increase.**

Both sentences are true. They simply speak of different levels.

What to take away: a growth rate means nothing unless the **base** is stated. Items that start from a small base always show dramatic growth rates. That is why section 4 put only the "structurally settled" points in the text and held back absolute figures.

Exercise 3 — Locating the bottleneck

Suppose the following three observations are reported at the same time.

- Leading-edge logic wafer capacity has room to spare.
- Lead times for accelerators exceed 40 weeks.
- HBM is pre-contracted through next year's volume.

Where is the bottleneck? And why would a policy proposal to "expand wafer capacity" have no effect?

▾ Solution

The bottleneck is **in a stage after the wafer.** If logic wafers are left over but finished products do not come out, the blockage lies after the wafer. The candidate the three observations point to directly is HBM supply. Advanced packaging (CoWoS and

similar), which binds HBM and the logic die into one package, is also a candidate, but these three observations alone cannot tell whether packaging is blocked too: a shortage of HBM alone would produce exactly the same three observations. On the packaging side, you would have to check CoWoS capacity utilization or the packaging backlog separately.

In a serial process, output is set by **the narrowest neck.** Widening a stage that has spare room (wafers) does not raise total output by one unit; it only piles inventory up in front of the neck. What needs to grow is the HBM line, and, if a packaging queue is confirmed, CoWoS capacity as well.

One step further: if the HBM line is expanded, what gets blocked next? HBM uses DRAM dies, so it competes with ordinary DRAM capacity; it needs the TSV process and thermocompression and MR-MUF stacking equipment (and, in the next generation, hybrid-bonding equipment as well), and that equipment is again tied to a few suppliers. **Bottlenecks are not removed; they move** (section 5). This structure is the subject of Session 6 and Session 7.

What to take away: when you read industry news, set "what is left over" beside "what is short," and the location of the bottleneck shows. Look at only one side and you will always get the wrong prescription.

Self-study appendix B — leads for the discussion questions

The discussion questions in section 7 have no right answers. But a reader working alone may not even know what to base an answer on, so only **the axes of the argument** are given. The conclusions are yours to draw.

QUESTION	WHERE IT DIVIDES	GROUNDS THAT LET YOU ANSWER EITHER WAY
1. Which of the three layers to invest in?	**Time horizon**	Over 5 years, design and people (fast payback); over 20 years, equipment and materials (irreplaceability compounds). Add a particular country's current position (for Korea, strong in memory and weak in equipment) and the answer changes again
2. Are in-house chips rational?	**Volume × control over the workload**	Quantified by the break-even formula N^* in Session 4. The key is that the break-even volume is surprisingly small (about 3,000–5,000 units); the real constraint is whether the workload survives the two to three years of development
3. Measurement or definition?	**The nature of the error**	If it is measurement error, it converges as the sample grows. If it is a difference in definition, it does not converge however precisely you measure. The cases in section 4 are of the second kind → statistically, it is not the **variance of the estimator but the estimand** that differs
4. Why does the bottleneck move?	**Investment lags**	When a bottleneck shows, capital rushes there; when it opens up after a two-to-three-year lead time, the next narrowest point shows. The movement is evidence of normal operation, not failure. From the demand side, growth in models and context passes the pressure along KV cache → HBM → package → server heat density → facility power (section 5 ②). Overlay the two directions, and the next bottleneck is the link, among those the pressure has reached, where capacity expands most slowly
5. Has the memory cycle changed?	**The nature of demand**	Training demand is investment-like (one-off capital spending); inference demand is operational (ongoing billing). The "it has changed" side argues that as the latter's share grows, demand flattens. The other side argues that the lead-time structure of capacity is unchanged — **even with flat demand, if supply comes in steps, the cycle remains**

Self-study appendix C — key terms

Only the terms that first appear in this session. They are used from Session 2 on.

TERM	ONE-LINE DEFINITION
Fabless	A semiconductor company that only designs and outsources production
Foundry	A plant that manufactures other companies' designs under contract
Design house (design service company)	A design service company that implements and verifies a fabless company's design physically for a particular foundry process and passes it on to manufacturing
IDM (integrated device manufacturer)	A semiconductor company that designs, manufactures and sells
OSAT (outsourced semiconductor assembly and test)	A contractor specializing in back-end work (packaging and test)
EDA (electronic design automation)	Software tools for chip design
HBM (high bandwidth memory)	Memory that stacks DRAM vertically to maximize bandwidth
Accelerator	Umbrella term for chips specialized for particular computations (GPU, TPU, NPU, etc.)

TERM	ONE-LINE DEFINITION
ASIC (application-specific integrated circuit)	A custom chip built for a fixed purpose
Training / inference	The process of building a model / the process of using a built model
Arithmetic intensity	Operations performed per byte moved from memory [FLOP/byte]
KV cache (key-value cache)	Memory that stores intermediate results of tokens already produced to avoid recomputing them
Quantization	Representing weights with fewer bits to save capacity and bandwidth
(Process) node	A process-generation name such as "3nm." A marketing label, not a physical dimension
Yield	The share of what is made that works correctly
CoWoS (chip-on-wafer-on-substrate)	TSMC's flagship 2.5D advanced packaging method
Hyperscaler	A very large cloud operator (Google, AWS, Microsoft, Meta, etc.)
Bullwhip effect	Demand fluctuations amplifying as they travel upstream in a supply chain
Single point of failure	A point whose stopping stops the whole

---

Self-study appendix D — self-assessment

If you can answer each item **in your own words,** you are ready for the next session. If two or more stop you, go back to the relevant section in the "Reading order" above.

☐ Given an article on "the AI chip market size," I can immediately name three things to check → sections 1 and 4

☐ I can draw the three layers of the value chain and explain why margins and irreplaceability diverge in each → section 2

☐ I can say in one sentence why export controls aim at manufacturing equipment as well as finished chips, and what equipment controls tie up that chip controls do not → sections 2 and 5

☐ Without looking at the table, I can name four or more ways in which the hardware that training and inference need differs → section 3

☐ I can explain **why "reading the whole set of weights to produce one token" makes memory the bottleneck** → section 3 ★

☐ I can say why a larger batch raises efficiency and why it cannot grow without limit → section 3 ★

☐ I can tell the three axes of supply-chain risk apart and explain why the remedy for each is different → section 5

☐ I can explain the movement of the bottleneck from EUV → CoWoS → HBM as "normal operation" → section 5

The two ★ items **must** be passed. All of Session 2 is a refinement of those two sentences.

Before the next session

Three things were named in this session but left undefined. Session 2 sets them up properly.

LEFT OPEN	IN SESSION 2
Arithmetic intensity I	Formally defined as the horizontal axis of the roofline model
"Ridge point of 300 FLOP/byte"	Derived as $I^* = P_{\text{peak}} / B$ and checked against real chips
"Below a batch of 300, you are bound by bandwidth"	Quantified with the decoding-ceiling formula

The question that carries over: if memory that cannot keep up makes a faster chip useless, **where exactly is that limit drawn?**

Sources

Introductory material chosen by the study curriculum (the "AI semiconductors" item)

CURRICULUM SUBTOPIC	LINK	HOW THIS BOOK USES IT
2024 technology trends	KDB Future Strategy Research Institute, "AI Semiconductor Technology and Industry Trends" (Aug 2024, in Korean; domestic research collection of the KDI Economic Information and Education Center)	AI chip technology trends seen from a state-owned bank's research institute. Background to section 2 (value chain, Korea's position)
2026 industry outlook	Samil PwC, "Global Semiconductor Industry Outlook 2026 — Semiconductor and beyond" (in Korean, PDF)	An industry outlook from an accounting firm. Used as case material for reading market figures in section 4 — readers are encouraged to check its definitions and counting basis themselves
Supply-chain risk	PwC Consulting Korea, "The Paradox of the Semiconductor Supercycle" (in Korean)	Direct source for the "paradox of the supercycle" discussion in section 5. The structure by which expansion in a boom creates the next slump

Primary sources added in edition 2.0 (checked 2026-10-08)

- TSMC, Value Chain Alliance — design houses in section 2
- SK hynix, prospectus for its U.S. listing (Form 424B4, 2026) — original on SEC EDGAR. By revenue in Q1 2026: first in HBM (56.4% share), second in DRAM including HBM (29.1%), second in NAND (18.5%). All cite IDC figures (checked against the original 2026-10-08)
- NVIDIA H100 product specifications · NVIDIA DGX H100/H200 User Guide, Table 3, power specifications · NVIDIA DGX B200 specifications — measurement boundaries in section 4
- LBNL, "Berkeley Lab Report Evaluates Increase in Electricity Demand from Data Centers" (2025-01-15) — section 6, misconception 6
- eCFR, 15 CFR §734.9 Foreign-Direct Product (FDP) Rules — section 5 ③

Verification status

- The three-layer value chain and the main firms in each layer — standard industry classification. Verified
- The nature of design houses and TSMC's VCA — confirmed on TSMC's official page (2026-10-08; quoted wording rechecked 2026-10-22). The earlier name "Value Chain Aggregator" appears as the page title in Internet Archive copies of 2019-07-30 and 2020-11-16, and "Value Chain Alliance" from the 2022-09-25 copy on (checked 2026-10-22). The member list changes over time and is not given in the text
- ASML as the sole supplier of EUV lithography tools — verified (several sources agree)
- The three-company HBM oligopoly and its leader (SK hynix) — reconfirmed with IDC figures for Q1 2026 by revenue (2026-10-08, SK hynix prospectus). The order of second and third was not confirmed from a primary source and is not given in the text. **Exact share figures vary widely across sources and are not asserted in the text**
- H100 SXM TDP "Up to 700W (configurable)", DGX H100 maximum input power 10.2 kW, DGX B200 maximum system power about 14.3 kW — checked against the manufacturers' original specifications (2026-10-08; the H100 wording rechecked 2026-10-22). 5.6 kW and 4.6 kW are calculated values
- U.S. data center electricity 58 TWh (2014), 176 TWh (2023, about 4.4%), 325–580 TWh (2028 estimate, 6.7–12%), more than doubling 2017–2023 and its main cause (AI servers) — checked against the LBNL press release (2026-10-08)
- The advanced computing FDP rule and the SME FDP rule in 15 CFR §734.9 — current eCFR text checked (version updated as of 2026-10-05, read 2026-10-08). That the items in paragraph (h) are advanced integrated circuits and computers such as ECCN 3A090 and 4A090, and that paragraph (k) covers manufacturing equipment such as 3B001, was rechecked against the 2026-10-01 version of the text through the eCFR API (2026-10-08)
- The shift in which inference accounts for most AI compute spending — consistent across several industry analyses, but the exact timing and ratio vary by source. Described only as a structural trend
- Absolute size of the AI accelerator market — **unconfirmed, wide variation. No specific figure is given in the text**

- Exact percentage shares by company — **unconfirmed, wide variation. Only the structural ranking is described**

SESSION 2

AI Chips — Compute Semiconductors

KEY TAKEAWAY

The first principle of accelerator design is not "make computation fast" but "move less data." Fetching a piece of data once from DRAM costs two orders of magnitude more energy than one multiplication. The differences among CPUs, GPUs, NPUs, TPUs and ASICs can all be explained along two axes: **(1) how far the overhead per instruction is amortized, and (2) how many times data, once fetched, is reused.** The **roofline model** built in this session is a shared tool that every later session uses again. The ridge point of current accelerators sits near 300 FLOP/byte, while the arithmetic intensity of LLM decoding is one or two orders of magnitude lower. That gap is the concern of the whole book.

PRIMARY SOURCES

Element Korea, "What Is an AI Chip?" (in Korean) · MSAP, "What Is an AI Accelerator?" (in Korean) · ScienceON JAKO202412457639314, "Research Trends in Domestic and International AI chips" (in Korean) · ScienceON JAKO202504350488400, "A Comprehensive Study of AI Semiconductors and Key Application Fields" (in Korean) · InitNode, "CPU vs GPU vs TPU vs NPU" · Williams et al., Roofline (CACM 2009) · Jouppi et al., TPU paper, arXiv:1704.04760 · NVIDIA H100 product specifications · NVIDIA Blackwell Architecture Technical Brief

Learning objectives

1. Explain why CPUs are inefficient for AI computation in terms of energy, not "because the clock is slow."
2. Tell apart how GPU tensor cores and TPU systolic arrays each gain their efficiency.
3. Build the roofline model and determine by calculation whether a given workload is compute-bound or memory-bound.
4. Weigh the break-even point of developing an in-house ASIC quantitatively.

0. Self-study guide — for readers working alone

> **THIS SESSION BUILDS THE TOOL FOR THE WHOLE BOOK**
>
> The one roofline inequality set up here is reused again and again in Session 3, Session 4, Session 6 and Session 7. You can read the other sessions once and move on, but **this one will not stay with you unless you calculate it by hand.** A calculator is all you need, and Exercise 1 takes 5 minutes.

What to bring from the previous session

FROM SESSION 1	IN THIS SESSION
"To produce one token, the whole set of weights is read"	With fp16 weights, this sentence becomes the approximation $I \approx B_{\text{sz}}$
"With a batch below 300, you are bound by bandwidth"	You confirm that the 300 comes from $I^* = P_{\text{peak}} / B$
"The layer that earns the most and the layer that cannot be replaced are not the same"	Here you see that in small-batch decoding the **bottleneck of performance** is bandwidth, not computation

Prerequisite check

CONCEPT	ONE-LINE ANSWER	CAN YOU SKIP IT?
Matrix multiplication (GEMM)	The operation that multiplies two matrices. An $n \times n$ matrix multiplication takes about $2n^3$ FLOP	❌ Required
Log-log plot	A plot with both axes on a logarithmic scale. Power-law relations show up as straight lines	Needed to read the figure in section 6
fp16 / INT8 / INT4	One number represented in 2 bytes / 1 byte / 0.5 byte	❌ Required
Systolic array	A structure that streams data through a grid and repeats multiply-accumulate operations	✅ The idea only
CUDA	The programming environment for NVIDIA GPUs and its ecosystem	✅ The name only

Reading order (120 minutes recommended)

STEP	SECTION	TIME	NOTE
1	Sections 1–5 (chip structure)	35 min	Quickly. For now this is background
2	**Section 6, the roofline**	**35 min**	**Read closely.** Copy the equations onto paper yourself
3	Self-study appendix, Exercises 1 and 2	20 min	Right after section 6. Put them off and they lose half their effect
4	Section 7, software lock-in	10 min	
5	Exercises 3 and 4 + self-assessment	20 min	

If section 6 is hard — a detour in three sentences

The roofline has a grand name, but its content is grade-school arithmetic.

1. To get work done you have to **bring the materials in** (bandwidth B) and **process them** (compute P_{peak}).
2. **Whichever of the two is slower sets the overall speed.** Hence the $\min$.
3. Which one is slower depends on "how long you process 1 kg of material" (I). If processing takes long, compute is the bottleneck; if it is over quickly, transport is.

$$P_{\text{attainable}}(I) = \min\left(P_{\text{peak}},\ B \times I\right)$$

That is all the equation says. The rest is plugging in real numbers.

---

1. Getting the intuition — computing is cheap, delivering data is expensive

A logistics analogy

Suppose machining one part in a factory takes 1 second. If fetching that part from the warehouse takes 3 minutes, adding more production lines will not raise output at all. There are only two ways to improve.

- **Move the warehouse closer** (= on-chip SRAM, HBM placed right next to the chip)
- **Do several jobs with each part you fetch** (= data reuse, tiling, larger batches)

Almost every idea in AI accelerator design is one of these two. Everything else is detail.

The shock in numbers

According to Horowitz's classic summary (ISSCC 2014, for a 45nm process), the energy gap between computation and memory access looks like this.

OPERATION	APPROXIMATE ENERGY
8-bit integer add	0.03 pJ
32-bit floating-point add	0.9 pJ
32-bit floating-point multiply	3.7 pJ
32-bit on-chip SRAM read (8 KB cache)	5 pJ
32-bit **DRAM read**	**640 pJ**

The top three rows (add and multiply) are taken directly from Figure 1.1.9 of Horowitz's paper. The two memory rows are the 32-bit access values of Han et al. (2015), who cite the same 45nm table. Horowitz's figure gives memory for 64-bit access: cache 10 pJ at 8 KB, 20 pJ at 32 KB and 100 pJ at 1 MB, and DRAM 1.3–2.6 nJ. The SRAM value of 5 pJ is half the 8 KB cache value, so it grows as the cache grows.

So **reading data once from DRAM costs about 170 times as much as one floating-point multiplication.** Halve Horowitz's DRAM range to get 32-bit values, and the ratio is about 180–350 times. As processes shrank, the energy of computation kept falling, but the energy of off-chip memory access fell much less, so the gap has widened over time.

> **The conclusion that follows:** the objective function a chip designer must optimize is not FLOPS but **"total energy per useful operation,"** and its dominant term is data movement.

---

2. Why the CPU will not do

First, clear away a misconception

"CPUs are slow because they have few cores" is only half right. The more fundamental reason is **the fixed cost the CPU pays per instruction.**

What actually happens when a CPU executes the single line `c = a * b`:

1. Instruction fetch (I-cache access)
2. Instruction decode
3. Branch prediction, register renaming, out-of-order scheduling
4. Register file read
5. **The actual multiplication** ← only this is "work"
6. Write-back of the result, commit

Steps 1–4 and 6 use far more energy and area than step 5. The CPU is designed on **the assumption that "it does not know what comes next,"** and the cost of preparing for that uncertainty is the overhead.

But deep learning does know what comes next

Matrix multiplication has no data-dependent branches, its access pattern is fixed statically, and the same operation repeats billions of times. **With no uncertainty, the whole cost of preparing for it can be removed.** That is the entire reason accelerators exist.

OVERHEAD ITEM	CPU	GPU	TPU/ASIC
Instruction fetch and decode	For scalars, every operation; with SIMD and AMX, amortized per vector or tile	Amortized per warp	Almost none (fixed dataflow)
Branch prediction	Present (expensive)	Minimal	None
Out-of-order execution	Present	None	None
Cache hierarchy	Large and complex	Small, can be controlled explicitly	Scratchpad (software-managed)
Share of area for control logic	Very large	Medium	Very small

Steps 1–6 above describe a single scalar instruction. Real CPUs also load several operations onto one instruction. A SIMD instruction processes a whole vector, and Intel AMX goes all the way to matrices. According to Intel's AMX code-sample article, AMX consists of tiles, which are two-dimensional registers, and TMUL, an accelerator that multiplies tiles as matrices. In the palette 1 configuration there are 8 tile registers totaling 8 KB, and one tile is at most 16 rows × 64 bytes. The same article names 4th Gen Xeon Scalable processors and the Xeon Max series as the processors it applies to. So the description of a CPU handling operations one at a time is an introductory shorthand. The conclusion does not change, though. The CPU still carries the control machinery it needs for generality, so it cannot amortize the same work as deeply as an accelerator.

In short: going from CPU to GPU to ASIC, you "sell flexibility to buy efficiency." The terms of that trade are the ASIC economics of section 5.

---

3. What a GPU really is — what one line of `matmul` does in hardware

The SIMT execution model

A GPU runs threads in bundles of 32 (a warp), and **the 32 threads of a warp execute the same instruction at the same time.** The cost of fetching and decoding an instruction is spread over 32 operations, so the overhead becomes 1/32.

Readers who have written GPU code will know the performance traps that come from this.

- **Warp divergence:** if threads in the same warp take different branches, the two paths run one after the other, and threads that do not belong to the path being run sit idle. If the threads split half and half between two paths of equal length, throughput halves; the size of the loss depends on the path lengths and the split ratio. → This is what the advice "avoid `if` on the GPU" really means.
- **Memory coalescing:** global memory is read in 32-byte transactions. If the 32 threads of a warp each read one consecutive 4-byte value, that is 128 bytes in all and takes four transactions; if the addresses are scattered 32 bytes or more apart, it takes one per thread, 32 in all. → This is why tensor layout (NCHW vs NHWC) decides performance.
- **Occupancy:** the GPU hides memory waits by running other warps, so the number of warps that can be resident at once matters. GPUs have L1 and L2 caches and shared memory too, but caches smaller than a CPU's cannot remove all of the latency, so **much of the latency is hidden by parallelism.**

Tensor cores — amortization taken to the limit

A tensor core performs the multiply-accumulate of small matrix blocks, $D = A \times B + C$, as one instruction. For a 16×16 block product, say, one instruction covers thousands of MACs, so the instruction overhead effectively goes to 0.

One line of `torch.matmul`, seen as a hierarchy:

```
torch.matmul(A, B)
  └─ cuBLAS/cuBLASLt kernel selection (tile size, split strategy)
       └─ grid/block decomposition: split the output matrix into tiles
            └─ load tiles from global memory (HBM) → shared memory (SRAM)
                 └─ shared memory → register fragments
                      └─ tensor core MMA instructions (block multiply-accumulate)
                           └─ accumulated results passed back up
```

The key point: this whole hierarchy exists for one purpose, "reuse each tile fetched from HBM as many times as possible." A block product with tile size T reads $O(T^2)$ bytes and performs $O(T^3)$ operations, so the arithmetic intensity grows as $O(T)$. **Tiling is the software way of raising the roofline's I.**

The precision ladder

AI accelerators gain throughput roughly by multiples as precision drops (FP32 → TF32 → BF16/FP16 → FP8 → FP4). There are two reasons.

1. **Arithmetic-unit area:** the area of a multiplier is roughly proportional to the square of its bit width, so halving the bits lets you fit about 4 times as many in the same area.
2. **Memory traffic:** half the data means half the bandwidth demand. **The real gain from quantization comes from bandwidth, not computation.** This is why quantization matters so much in inference.

---

4. TPUs and systolic arrays — building reuse into the hardware

Structure

A systolic array lays out processing elements (PEs) in a grid and streams data through it, with **each PE passing values on to its neighbor.** The name comes from the way data flows in a regular rhythm, like blood pushed out by the heart (systole).



Activations flow left to right; each PE multiply-accumulates and passes them on to its neighbor — data read once is reused inside the array

In an $N \times N$ array, one input element is reused N times as it passes through the array. That is, each piece of data read once from memory is used for $O(N)$ operations. **The arithmetic intensity is guaranteed by the wiring, not by software.**

Google's TPU v1 arranged $256 \times 256 = 65{,}536$ 8-bit MAC units this way. Data moving inside the array travels over short wires between neighboring PEs, so the energy is extremely low.

The figure below shows the fourth beat when one token ($M = 1$, decoding) is streamed through a 4×4 array whose weights sit in the PEs. In the screen edition you can step through the beats one at a time, or change the number of tokens M and the matrix size, and count how many PEs are working at each beat.

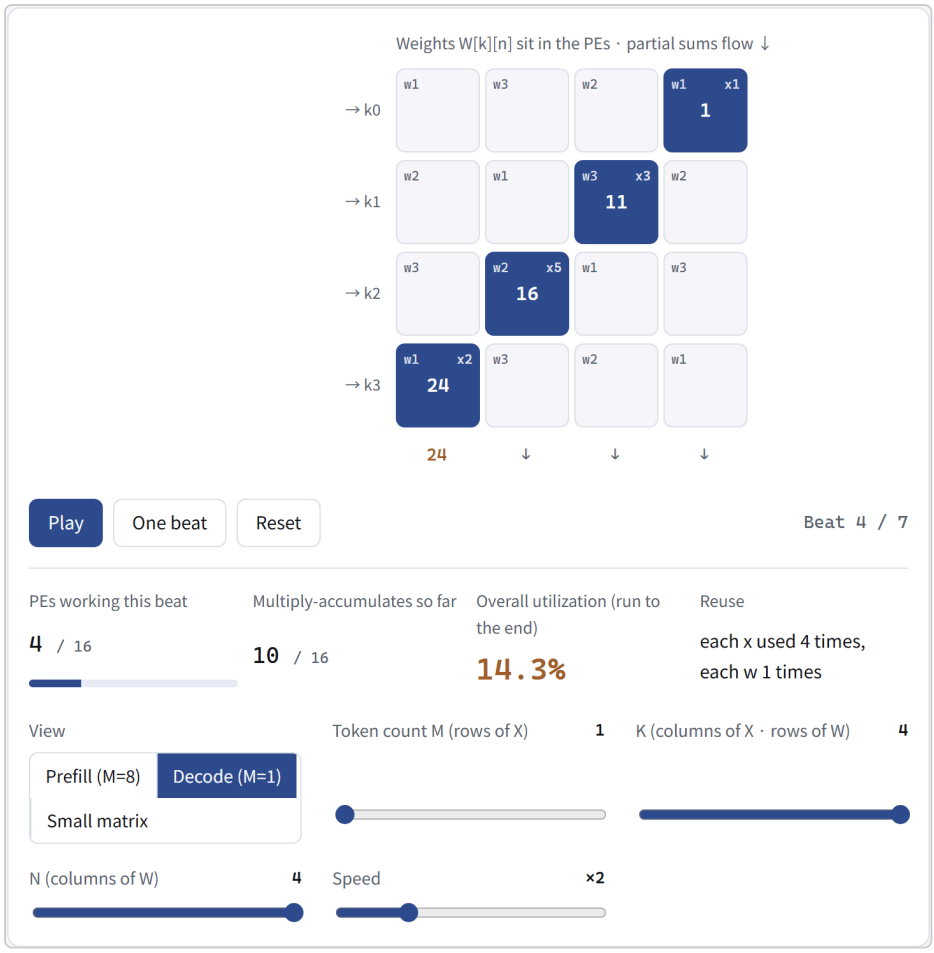


Figure 2-1. The fourth beat of one token (M = 1, batch-1 decode) flowing through a 4×4 weight-stationary systolic array. While the activations pass along the diagonal, only the 4 PEs on the diagonal out of 16 are working. — In the HTML edition you can change the values here (HTML edition, Session 2, Section 4).

GPU vs. systolic array — what was given up and what was gained

	GPU	SYSTOLIC ARRAY
Guarantee of reuse	Software (tiling, kernels) is responsible	Guaranteed by the hardware structure
Flexibility	Can run arbitrary kernels	Best for matrix-multiplication shapes, inefficient for anything else
Programming	Mature ecosystem such as CUDA	Very heavy dependence on the compiler
Utilization	Depends heavily on kernel quality	Drops sharply when the matrix size does not fit the array

The weakness of the systolic array: if the matrix size is not a multiple of the array dimension, the PEs at the edges sit idle. When the batch is small or the matrix is long and thin (decoding!), utilization falls a long way. This is why designs dedicated to inference take a different form from those for training.

Counted in beats, it looks like this. With the weights fixed in the PEs, streaming M tokens through one $K \times N$ tile takes $M + K + N - 2$ beats to fill and drain. At TPU v1 size ($K = N = 256$), with $M = 1$ the PEs are working only 0.2% of the time on average over 511 beats; with $M = 4{,}096$ that rises to about 89%. Real chips feed the next tile in right behind

the current one to overlap the fill and drain time, and loading the weights into the PEs takes time of its own. Still, when M is small each weight is used only M times and then discarded, so overlapping gains little. There are also several dataflows besides weight-stationary, such as output-stationary (the textbook by Sze et al. in section 11).

---

5. NPUs and ASICs — the economics of specialization

NPU

A neural-network accelerator for mobile and edge devices. Its objective function is different.

- What it optimizes is not absolute performance but **TOPS/W** (performance per watt)
- Centered on low precision (INT8/INT4), under battery and heat constraints
- Integrated in an SoC with the CPU, GPU and ISP, sharing memory (LPDDR)
- Use is intermittent, so latency and wake-up overhead matter

The fundamental difference from server accelerators: a server spends power to get throughput, whereas at the edge the power budget is fixed and you squeeze out the most performance inside it. Both are "AI chips," but their design goals are close to opposite.

Calculating the ASIC break-even point (exercise)

Whether to build a dedicated chip or buy one can be quantified.

- C_{NRE} : non-recurring engineering cost (design staff, EDA licenses, IP licenses, **mask set**, verification and prototypes). At leading-edge nodes the mask set alone runs to tens of millions of dollars, and total NRE can reach hundreds of millions of dollars.
- c_A : ASIC unit cost (wafer cost ÷ number of good dies + packaging + test)
- c_G : unit acquisition cost of the alternative (a commercial GPU)
- N : expected volume

Here we assume that one ASIC does the work of one GPU at the same throughput, latency and quality (1:1 replacement). Only then does the same volume N mean the same demand.

Comparing total costs: $C_{\text{NRE}} + Nc_A \ < \ Nc_G$

$$N^* = \frac{C_{\text{NRE}}}{c_G - c_A}$$

If the ASIC's throughput is r times the GPU's, the same demand needs N / r ASICs. In that case put c_A / r in place of c_A (if $r < 1$, the break-even volume grows).

But this formula is only half right. In practice three more terms have to go in.

1. **The time axis:** developing a chip takes 2–3 years, and model architectures change faster than that. When the chip is finished, will that workload still dominate?

2. **Electricity:** in a data center, several years of electricity bills can exceed a chip's acquisition price in its total cost of ownership (TCO). Put operating costs into c and the dedicated chip looks far better. Counted per unit, however, what decides the break-even count is mostly the difference in acquisition price; the difference in electricity costs is only a few percent (solution to Exercise 4). Electricity becomes a large sum only in totals at the scale of tens of thousands of units running for several years.
3. **Bargaining power:** having your own chip gives you option value in price negotiations with commercial GPU vendors. Even if you never use it, its existence is worth something.

> **Try it yourself:** with these three terms added, explain "why hyperscalers build their own chips and most startups do not." The heart of the answer is the difference in scale of N and C_{NRE}, and whether the workload is stable.

The flexibility-efficiency spectrum



The FPGA is an interesting middle point. Its hardware can be reconfigured, so it needs no mask set, but it implements the same function with far more area and power than an ASIC. It suits low-volume, high-mix products and early stages when the algorithm is still in flux.

---

6. The roofline model — the shared tool of this book

Definition

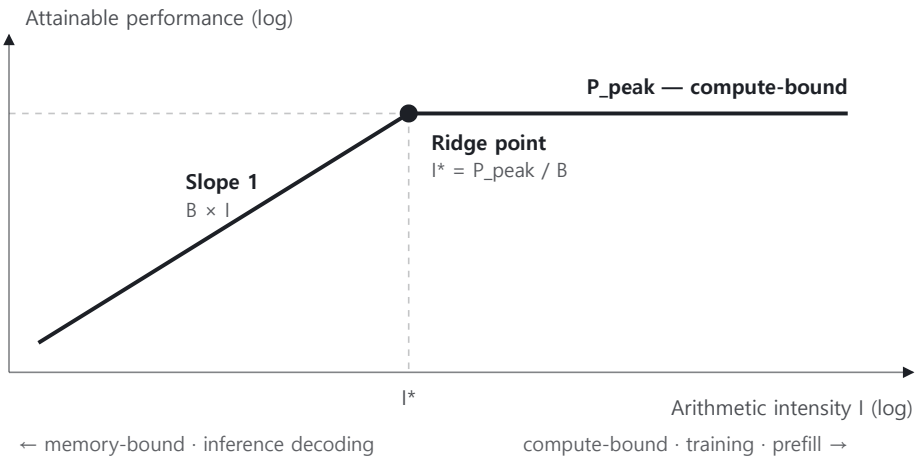
The performance ceiling is the minimum of two constraints.

$$P_{\text{attainable}}(I) = \min\left(P_{\text{peak}},\ B \times I\right)$$

- P_{peak} : the chip's peak compute rate [FLOP/s]
- B : memory bandwidth [byte/s]
- I : arithmetic intensity [FLOP/byte] = (operations performed) / (bytes moved from memory)

Plotted on log-log axes, a line of slope 1 (the bandwidth limit) meets a horizontal line (the compute limit) to form the shape of a roof. The point where it bends is the **ridge point**,

$I^* = P_{\text{peak}} / B.$



Ridge points of real chips

CHIP	P_{PEAK} (BF16 DENSE)	B	I^*
NVIDIA H100 SXM	989 TFLOP/s	3.35 TB/s (HBM3, 80 GB)	≈ 295 FLOP/byte
NVIDIA B200	2,250 TFLOP/s	7.7 TB/s (HBM3E; chip's published maximum 192 GB · 180 GB per GPU in the DGX B200)	≈ 292 FLOP/byte

> **Caution:** vendor material also lists a figure twice as high with structured sparsity. The values above are dense. Always check which one you are looking at on a spec sheet.
>
> **Capacity also differs between chip and system.** NVIDIA's Blackwell Architecture Technical Brief (V2.1) gives "Up to 192 GB" as the GPU memory of the HGX B200. The user guide and datasheet for the DGX B200, the finished server, give 1,440 GB in total for 8 GPUs, which works out to 180 GB per GPU. When you plug capacity into a calculation, state whether it is the chip's published value or a system specification. Bandwidth also varies by document. Technical Brief V2.1 gives the bandwidth of the same HGX B200 GPU as 7.7 TB/s, and the DGX B200 datasheet gives 64 TB/s in total for 8 GPUs (8.0 TB/s per GPU by calculation). The tables and figures in this book use the chip values from the Technical Brief (192 GB, 7.7 TB/s), the same source as the capacity. Calculated with 8.0 TB/s, $I^* \approx 281$, so the conclusion below, that it sits near 300, holds either way. The compute figure of 2,250 TFLOP/s is half of the sparse 4.5 PFLOPS in the same table (the table's dense column gives 2.2).

What to notice: from one generation to the next, I^* barely moves from around 300. When compute grew 2.3 times, bandwidth was also raised 2.3 times, which means the balance was **only just kept.** Getting that bandwidth took enormous spending, from HBM3 to HBM3E and on to HBM4. That is the subject of Session 3.

Placing LLM workloads on the roofline

(a) Training / prefill

The tokens of sequence length S and batch B_{sz} are processed all at once, so this becomes a large GEMM. When a weight matrix $W \in \mathbb{R}^{n \times n}$ is applied to M tokens:

- Computation $\approx 2Mn^2$ FLOP
- Memory traffic $\approx 2n^2$ bytes (fp16 weights loaded once)
- Arithmetic intensity $I \approx M$ (when activation traffic is left out of the count)

Leaving out the activations is an approximation. If the $M \times n$ input is read from HBM and an output of the same size is written back, the traffic becomes $2n^2 + 4Mn$ bytes and $I = Mn \,/\, (n + 2M)$. This value cannot exceed the smaller of M and $n \,/\, 2$. So $I \approx M$ holds when n is much larger than M, or when kernel fusion keeps the activations on chip. In LLMs n is in the thousands or more ($n = 8{,}192$ and $M = 4{,}096$ give $I = 2{,}048$), so whichever way you count, $I \gg I^{*}$ → **compute-bound.** The GPU's compute units are well filled.

(b) Inference / decode

Tokens are generated one at a time. Even if a batch of B_{sz} sequences is processed together, each step handles one token per sequence.

- Computation $\approx 2B_{sz}n^2$ FLOP
- Memory traffic $\approx 2n^2$ bytes (the weights must be read again in full)
- Arithmetic intensity $I \approx B_{sz}$

With fp16 weights, a batch below 300 is memory-bound. If you serve with a batch of 32, $I \approx 32$, so you use only about $32 \,/\, 295 \approx 11\%$ of the theoretical performance. The other 89% of the compute units sit idle waiting for memory.

The figure below shows where the workload falls under the roof when decoding runs on an H100 with fp16 weights and a batch of 32. In the screen edition you can change the chip, phase, batch, weight precision and model size and see when the point crosses the ridge point.



Figure 2-2. Decode workload on the roofline. On an H100 at batch size 32 with fp16 weights, arithmetic intensity is about 32, short of the ridge point (about 295), so the workload is memory-bound and the compute-unit utilization ceiling is about 11%. — In the HTML edition you can change the values here (HTML edition, Session 2, Section 6).

The ceiling on decoding speed — a rule of thumb counted in bandwidth

For a model with P weights (parameters) at b bytes per parameter, the maximum number of tokens generated per second is set by bandwidth alone.

$$\text{tokens/s} \lesssim \frac{B}{P \cdot b} \times B_{\text{sz}}$$

Example: to run a 70B model in fp16 ($b = 2$) on one H100 with a batch of 1, the weights alone are 140 GB and do not fit in 80 GB. Split across 2 cards, the total bandwidth is 6.7 TB/s, so the theoretical ceiling is 6.7e12 / (1.4e11) ≈ 48 tokens/s. This rule of thumb assumes that when the weights are spread over N chips, each chip reads its share at the same time, so the bandwidth adds up N times. Quantized to INT8 ($b = 1$), the weights are 70 GB and only just fit on one card. Drop to one card and the bandwidth also falls to 3.35 TB/s, so the ceiling stays at about 48 tokens/s; keep both cards and it doubles to about 96 tokens/s.

This formula counts only the bandwidth side, so it holds only when the arithmetic intensity I is below the ridge point I^*. If only the weights are read at b bytes and the computation runs at the BF16 peak, then $I = 2B_{\text{sz}} / b$, and the batch at which you cross into compute-bound is $B_{\text{sz}} = (b / 2)\, I^*$. For the H100 that is about 295 in fp16, about 148 in INT8 and about 74 in INT4. If the computation also runs at the same low precision and the peak rises with it, the

boundary moves back to around the BF16 I^* (solution to Exercise 3). When compute-bound, the ceiling is set by the $2P$ FLOP that one token takes (for the whole batch, tokens/s $\lesssim NP_{\text{peak}} / (2P)$, where P_{peak} is the peak at the precision actually used). The traffic from reading the KV cache at every step is also left out, so the longer the context, the further real speed falls below this ceiling. Translated into a metric users feel, for a single sequence this formula gives a lower bound $P \cdot b / B$ on the time between output tokens (TPOT). The time to the first token (TTFT) is set not by this formula but by prefill ("Conditions for a fair comparison" at the end of section 8).

> **What this one rule of thumb explains:** why quantization is essential in inference, why you try to grow the batch, why "do the weights all fit in memory?" is decisive, and why the HBF (high-capacity memory) of Session 3 is being discussed.

Every way to raise arithmetic intensity (summary)

METHOD	WHERE	HOW IT RAISES I
Tiling / blocking	Software	Reuses tiles loaded on chip many times
Kernel fusion (e.g., FlashAttention)	Software	Keeps intermediate results on chip instead of writing them to HBM
Larger batches	Serving policy	Processes more sequences with the same weights
Quantization	Model	Fewer bytes moved, so a smaller denominator
MoE / sparsity	Model structure	Not a way to raise I. Only the weights one token uses are read and computed, so computation and traffic fall together and I stays the same; fewer bytes per token raise the decoding ceiling. With MoE, when the tokens in a batch scatter across many experts, fewer tokens reuse each expert and I actually falls (and routing overhead is added)
Systolic array	Hardware	Reuse guaranteed by structure
Large on-chip SRAM	Hardware	Cuts the number of off-chip accesses itself
More HBM bandwidth	Hardware	Raises B, not I (→ Session 3)
PIM	Hardware	Moves computation toward the memory (→ Session 3)

Worth pinning down: in this list, the software items all raise I, and half of the hardware items raise B. **Once you see the picture of two camps attacking different terms of the same inequality, papers and articles become much easier to read.**

---

7. Software lock-in — a moat outside the technology

Even when hardware performance is similar, software decides what actually gets adopted.

- **The CUDA ecosystem:** optimized kernels built up over many years, such as cuDNN,

cuBLAS, NCCL and TensorRT, and extensive library support.

- **The alternatives:** AMD ROCm, Intel oneAPI, Google XLA/JAX, OpenAI Triton, the compiler stack of PyTorch 2.x. Attempts like Triton to describe kernels in a hardware-neutral way could weaken lock-in. A compiler stack captures the model as a computation graph, lifts it into an intermediate representation (IR), and then lowers it step by step into representations closer to the hardware to produce kernels. According to the PyTorch 2.14 documentation, in `torch.compile` TorchDynamo captures the graph from Python code, and the default compiler, TorchInductor, generates code for a range of accelerators and backends. On NVIDIA, AMD and Intel GPUs, TorchInductor builds its kernels on OpenAI Triton.
- **Why hyperscalers can build their own chips:** they control their own workloads, so they only have to fit the compiler stack to their own models. **Not needing to build a general-purpose software ecosystem is their decisive advantage.**

Operator coverage and fallback

Here is a common reason why the hardware is good but the model is slow. Accelerator libraries do not cover every operator. When the runtime meets an operator it does not support, it hands just that node to the CPU (fallback). The input then has to be moved down from device memory to host memory, and the result the CPU computes has to be moved back up to the device; before and after each transfer, device and host wait for each other. Even if the operation itself is small, this round trip adds to the latency of the whole graph.

Two widely used runtimes document this behavior. ONNX Runtime takes a list of execution providers in order of priority. Given `['CUDAExecutionProvider', 'CPUExecutionProvider']`, it runs on CUDA the nodes CUDA can handle and runs the rest on the CPU. PyTorch's Apple GPU backend (MPS) sends operations that MPS does not support to the CPU when the environment variable `PYTORCH_ENABLE_MPS_FALLBACK` is set to 1 (PyTorch 2.14 documentation).

How expensive the round trip is becomes clear when you set the widths of the two paths side by side. The NVIDIA H100 spec sheet lists PCIe Gen5 for the H100 SXM at 128 GB/s and GPU memory bandwidth at 3.35 TB/s. The H100 white paper states that 128 GB/s is the total for both directions, 64 GB/s each way. Dividing the official figures gives about 26 times; counting one direction, about 52 times. A single fallback moves data that would have stayed inside HBM twice through a path more than an order of magnitude narrower. The claim that lock-in really consists of accumulated kernels (appendix B, question 4) takes a concrete shape here. A new chip with a short operator list can be slowed by fallback even if its arithmetic units are fast.

Case — DeepSeek-V3, hiding communication behind computation

The DeepSeek-V3 technical report (arXiv:2412.19437) gives an example of software making up for hardware constraints. According to the report, the model was trained on a cluster of 2,048 H800s, and full training (2,664K GPU hours for pre-training, 119K for context-length extension, 5K for post-training) took 2.788M H800 GPU hours. The report assumes a rental price of USD 2 per GPU hour to arrive at USD 5.576 million, and in the same paragraph states that this figure covers only the official training and excludes the costs of prior research and ablation experiments on architecture, algorithms and data. So the figure must not be read as the cost of developing the model.

What matters technically is how communication was handled. The report designed DualPipe, an algorithm that overlaps the computation and communication phases of the forward and backward passes in pipeline parallelism, and set aside 20 of the H800's 132 SMs to handle cross-node expert (MoE) communication. That works out to about 15% of the compute units. The report itself notes that this approach limits computational throughput, and it suggests to chip designers that communication be moved off the SMs onto dedicated hardware. It is a case of hiding cross-node communication, an item outside the roofline (appendix B, question 1), behind computation by scheduling, and of giving up part of the compute units in return.

---

8. Common misconceptions

Misconception 1: "High FLOPS means a fast chip" → If the workload's arithmetic intensity is below the ridge point, FLOPS mean nothing. In LLM inference, the bandwidth figure is the better predictor.

Misconception 2: "You can compare TOPS/TFLOPS figures as they are" → They differ by 4–8 times depending on precision (FP32/BF16/FP8/INT8/INT4) and on whether sparsity is applied. Match the conditions before comparing. Even matching precision and sparsity is not enough — see "Conditions for a fair comparison" at the end of this section.

Misconception 3: "GPU utilization at 90% means it is well used" → The utilization reported by `nvidia-smi` is "the share of time a kernel is running," not a sign that the compute units are full. A kernel spinning while it waits for memory also shows as 100%. The real metric is achieved FLOPS against theoretical FLOPS, such as **MFU (Model FLOPs Utilization).** When kernels are very short there is a trap on the other side, too. The overhead of launching a kernel and waiting for it becomes larger than its run time, and the GPU sits idle between kernels (the CUDA Graphs example in appendix B, question 1).

Misconception 4: "Quantization makes computation faster" → The throughput of the arithmetic units does rise, but in inference most of the gain you notice comes from **less memory traffic.**

Misconception 5: "An ASIC is always more efficient than a GPU" → Only when the workload is fixed. When matrices that do not fit the array size or new operators (for example, a new attention variant) come in, utilization drops sharply.

Misconception 6: "NPUs and GPUs are the same thing at different sizes" → Their objective functions differ (throughput vs. performance per watt). There are many points where design decisions go in opposite directions.

Conditions for a fair comparison — what to match besides precision

The speed of an inference service cannot be captured in one number. The latency a user feels splits into three. **TTFT** (time to first token) is the time from sending the request until the first token comes out, and it is governed by prefill, which processes the whole input at once. **TPOT** (time per output token) is the average interval between output tokens, governed by the decoding ceiling of section 6. **P99** is the tail latency below which 99% of requests finish. Even with a good average, a long tail means the service-quality promise cannot be kept.

That is why MLPerf Inference, the industry-standard benchmark, ties several conditions together when it compares chips. In the data-center Llama2-70B task of the MLCommons inference rules (the master version checked on October 8, 2026, last modified August 20, 2026), the Closed division requires the same model and preprocessing as the reference implementation (quantization calibration is allowed, retraining is not) and fixes the dataset as OpenOrca. Accuracy must be at least 99.9% of the FP32 reference ROUGE scores, and the latency constraints are TTFT 2,000 ms and TPOT 200 ms in the Conversational category, TTFT 450 ms and TPOT 40 ms in the Interactive category. In the Server and Interactive scenarios, where requests arrive at random following a Poisson distribution, latency is measured at the 99th percentile. A chip that raised its throughput with lower precision also has to clear the accuracy floor and the latency constraints for its result to count. One line of a TOPS table tells you none of this. The rules change with every round, so when you cite them, state the version.

---

9. Quantitative exercises

Exercise 1 — Computing the ridge point Pick any accelerator from a spec sheet, find the dense P_{peak} and B, and compute I^*. Compare three generations and look at the trend in I^*.

Exercise 2 — The decoding ceiling For 7B / 70B / 405B models served in fp16, INT8 and INT4, compute the theoretical maximum tokens/s at a batch of 1 on a given accelerator. Which combinations meet a real service latency requirement (for example, 30 tokens/s)?

Exercise 3 — The KV cache budget With sequence length S, layers L, total KV head dimension d (number of heads storing K and V × dimension per head; query heads are not counted) and batch B_{sz}, the size of an fp16 KV cache is roughly $2 \cdot L \cdot S \cdot d \cdot B_{\text{sz}} \cdot 2$ bytes. Find the largest batch possible for a given memory capacity, and place the arithmetic intensity at

that batch on the roofline. **How the capacity constraint limits bandwidth efficiency** comes out.

Exercise 4 — ASIC break-even Using assumed values (NRE USD 100 million, ASIC unit price USD 3,000, GPU unit price USD 30,000, 3 years of electricity included), find N^* and discuss whether it is achievable if the model's life is 2 years.

---

10. Discussion questions

1. What does the roofline model fail to explain? (Latency, the cache hierarchy, communication, kernel launch overhead and so on. Knowing a model's limits is half of knowing the model.)
2. Is it chance or design that the ridge point stays near 300 across generations? What happens if it moves away in either direction?
3. What effect does MoE (mixture of experts) have from the roofline's point of view? How should we assess a structure in which total parameters grow but active parameters shrink?
4. How long can software lock-in last? What changes as intermediate layers such as Triton and XLA mature?
5. If you were designing a new chip only for inference, what would you do differently from a training chip? (Hint: where would you put the ridge point?)

---

11. Further study

- **The roofline original:** Williams, Waterman & Patterson, "Roofline: An Insightful Visual Performance Model for Multicore Architectures", *CACM* 52(4), 2009 — DOI 10.1145/1498765.1498785
- **The source of the energy figures:** Horowitz, "Computing's Energy Problem (and what we can do about it)", ISSCC 2014 keynote — DOI 10.1109/ISSCC.2014.6757323. The two 32-bit memory values in the section 1 table come from Figure 1 of Han, Pool, Tran & Dally, "Learning both Weights and Connections for Efficient Neural Networks", NIPS 2015 — arXiv:1506.02626
- **TPU architecture:** Jouppi et al., "In-Datacenter Performance Analysis of a Tensor Processing Unit", ISCA 2017 — the paper that sets out the design rationale of TPU v1 directly. **If you read only one paper alongside this session, make it this one.** arXiv:1704.04760
- **Accelerator design textbook:** Sze, Chen, Yang & Emer, "Efficient Processing of Deep Neural Networks" (Synthesis Lectures, 2020) — the standard treatment of the dataflow taxonomy (weight stationary / output stationary and so on). DOI 10.1007/978-3-031-01766-7

- **Kernel-level optimization:** Dao et al., the "FlashAttention" series — the leading example of raising I through kernel fusion. The first paper is arXiv:2205.14135
- **GPU architecture:** NVIDIA architecture white papers (H100 white paper, Blackwell Architecture Technical Brief) and the memory-hierarchy chapters of the CUDA Programming Guide
- **Kernel launch overhead:** NVIDIA technical blog, "Getting Started with CUDA Graphs" (2019) — an example that measures how batching short kernels into one launch works and what it gains
- **The rules of comparison:** MLCommons MLPerf Inference rules — accuracy floors and TTFT/TPOT constraints for each task

---

Self-study appendix A — solutions to the quantitative exercises

These are the solutions to the exercises in section 9. **Solve them yourself first, then open them.** The numbers come from public spec sheets, and all are dense (no sparsity) values.

Solution to Exercise 1 — Computing the ridge point

▾ Solution

Plug three generations into $I^* = P_{\text{peak}} / B$.

CHIP	P_{PEAK} (BF16 DENSE)	B	I^*
A100 80GB SXM	312 TFLOP/s	2.04 TB/s	**153** FLOP/byte
H100 SXM	989 TFLOP/s	3.35 TB/s	**295** FLOP/byte
B200	2,250 TFLOP/s	7.7 TB/s	**292** FLOP/byte

Calculation: 312 / 2.04 = 153, 989 / 3.35 = 295, 2250 / 7.7 = 292.

Seen as growth rates from generation to generation, it is much clearer.

STEP	COMPUTE	BANDWIDTH	I^*
A100 → H100	×3.17	×1.64	×1.93 (worse)
H100 → B200	×2.28	×2.30	×0.99 (almost unchanged)

How to read it: in the A100→H100 generation, compute grew much faster than bandwidth, and the ridge point nearly doubled. That is, **it became easier to end up memory-bound.** From H100 to B200, HBM3E was attached to raise bandwidth at almost the same rate as compute, and the ridge point stayed where it was.

Implication: I^* staying near 300 is no accident. It **means bandwidth is only just keeping up with compute**, and the price of that "only just" is the cost of HBM and advanced packaging. The reason Session 3 and Session 6 exist is contained in this one

table.

> **A caution when reading spec sheets:** vendor material also lists values with sparsity applied, and using them inflates I^* twofold. Always check that a figure is dense.

Solution to Exercise 2 — The decoding ceiling

▾ Solution

$$\text{tokens/s} \lesssim \frac{B}{P \cdot b} \times B_{sz}$$

The calculation is for one H100 SXM ($B = 3.35$ TB/s, capacity 80 GB) at a batch of 1. $P \cdot b$ is the total number of bytes in the weights.

MODEL	PRECISION	WEIGHT SIZE	CARDS NEEDED	TOTAL B	THEORETICAL CEILING
7B	fp16 ($b = 2$)	14 GB	1	3.35 TB/s	**239 tok/s**
7B	INT8	7 GB	1	3.35	479
7B	INT4	3.5 GB	1	3.35	957
70B	fp16	140 GB	**2** (exceeds capacity)	6.7	**48 tok/s**
70B	INT8	70 GB	1 (tight)	3.35	48
70B	INT4	35 GB	1	3.35	96
405B	fp16	810 GB	**11**	36.85	45
405B	INT8	405 GB	6	20.1	50
405B	INT4	202.5 GB	3	10.05	50

Worked example: 70B fp16 on 2 cards → 6.7×10^{12} / $(1.4 \times 10^{11}) = 47.9$ tok/s. Total B is the number of cards × 3.35 TB/s, and the ceilings were computed from unrounded values.

Combinations that meet the 30 tok/s requirement: in the table above, all of them. But note that these are **theoretical ceilings.** In practice, bandwidth utilization reaches only 60–80% (an assumed value in this book), and tensor-parallel communication and kernel launch overhead come on top. Utilization alone takes 70B fp16 on 2 cards from 48 to about 29–38 tok/s, and with overhead added it can fall below 30, which makes it a close call.

Three insights

1. **On the same equipment, model size and speed are inversely proportional.** P sits in the denominator of the formula, so with the number of cards and the precision fixed, the ceiling drops 10 times going from 7B to 70B. In the table above, fp16 7B (1 card, 239) and 70B (2 cards, 48) differ by 5 times because 70B uses twice as many cards and so has twice the total bandwidth. The decision "let's use a bigger model" is a decision to run that much slower on the same equipment, or to buy that many more chips for the same speed.
2. **Quantization wins twice.** Each time the weight bytes are halved, the ceiling doubles on the same number of cards, or the same ceiling is reached with half the cards. Switching 70B from fp16 to INT4 gets you both. 48 tok/s on 2 cards becomes 96 tok/s on 1, and the equipment cost halves.
3. **If you load a model onto just enough cards to hold it, the batch-1 ceiling becomes almost independent of model size.** 70B fp16 on 2 cards (48), 70B INT8 on 1 (48), 405B fp16 on 11 (45) and 405B INT4 on 3 (50) are all similar. Each chip reads, once per token, weights that nearly fill its own memory, so the ceiling clusters just above one chip's "capacity ÷ bandwidth" (for the H100, 80 GB ÷ 3.35 TB/s ≈ 24 ms, about 42 tok/s). For the same model and precision, adding chips raises this rule of thumb's ceiling in proportion to the number of cards (405B fp16, 11 cards 45 → 12 cards 50). In reality, tensor-parallel communication between chips is added, so it does not rise that much. This communication cost lies outside the rule of thumb (appendix B, question 1).

Solution to Exercise 3 — The KV cache budget

▾ Solution

The size of an fp16 KV cache is $4 \cdot L \cdot S \cdot d$ bytes per sequence (×2 for K and V, ×2 for fp16).

Setup: a 70B-class model, $L = 80$ layers, with GQA $d = 8$ heads $\times$ 128 $= 1{,}024$, $S = 4{,}096$, one H100 80GB.

KV per sequence $= 4 \times 80 \times 4096 \times 1024 = 1.34 \times 10^9$ bytes = **1.34 GB**

PRECISION	WEIGHTS	REMAINING CAPACITY	LARGEST BATCH	ARITHMETIC INTENSITY $I = 2B_{SZ} / b$	I^* AT THAT PRECISION	UTILIZATION I / I^*
INT8	70 GB	10 GB	**7**	14	590	**2.4%**
INT4	35 GB	45 GB	**33**	132	1,181	**11.2%**

(I^* here takes the H100 BF16 dense value of 295 and assumes the compute peak doubles for INT8 and quadruples for INT4. The doubling for INT8 matches the INT8 tensor core figure on the H100 spec sheet; the quadrupling for INT4 is an assumption

that extends the same ratio. In practice, activation buffers and fragmentation leave even less room, so these values are optimistic ceilings. On the other hand, assuming every sequence fills up to S is a conservative assumption. With schemes such as PagedAttention, which reserve space only as it is used, the average length is what counts.)

There are two precision assumptions. The table above is an approximation that assumes the computation runs at the same precision as the weights. Then arithmetic intensity and the ridge point both grow by a factor of $2 / b$, and utilization comes out as $B_{\mathrm{SZ}} / 295$, just as in fp16 (this is the "B_{SZ} / I^{*}" used by the KV figure for Exercise 3). The roofline figure of section 6 (screen edition) is the case where only the weights are read at low precision and the computation runs in BF16. Arithmetic intensity is the same $2B_{\mathrm{SZ}} / b$, but the ridge point is set at the BF16 value of 295, so utilization comes out $2 / b$ times higher. Calculated, the INT8 batch of 7 in the table above becomes about 4.7% and the INT4 batch of 33 about 45%. The two values differ not because of an error but because the assumptions differ.

What comes out — the heart of this exercise

The capacity constraint holds down the batch, a held-down batch holds down arithmetic intensity, and held-down arithmetic intensity leaves **97% of the compute units, the most expensive part, idle.** The bandwidth constraint and the capacity constraint look like separate things, but capacity governs bandwidth efficiency through the batch.

One step further — without GQA

With Multi-Head Attention, $d = 64 \times 128 = 8{,}192$, 8 times as large, and the KV per sequence is **10.7 GB.** In INT8, not even a batch of 1 fits. **Why GQA and MQA are a necessity, not an option,** lies in this one-line calculation.

Priorities among the means of raising throughput (they follow naturally from this calculation)

1. KV cache quantization (INT8) → KV halves, batch doubles
2. Weight quantization → remaining capacity jumps, batch jumps
3. Removing fragmentation with PagedAttention and similar → recovers effective capacity
4. Limiting sequence length → linear in S

The figure below shows how memory is divided when a 70B-class model is loaded onto one H100 80GB with INT8 weights, an fp16 KV cache and GQA, at $S = 4{,}096$. In the screen edition you can change the chip, sequence length, weight and KV cache precision, attention scheme and reserve ratio, and watch how the largest batch and compute-unit utilization move.

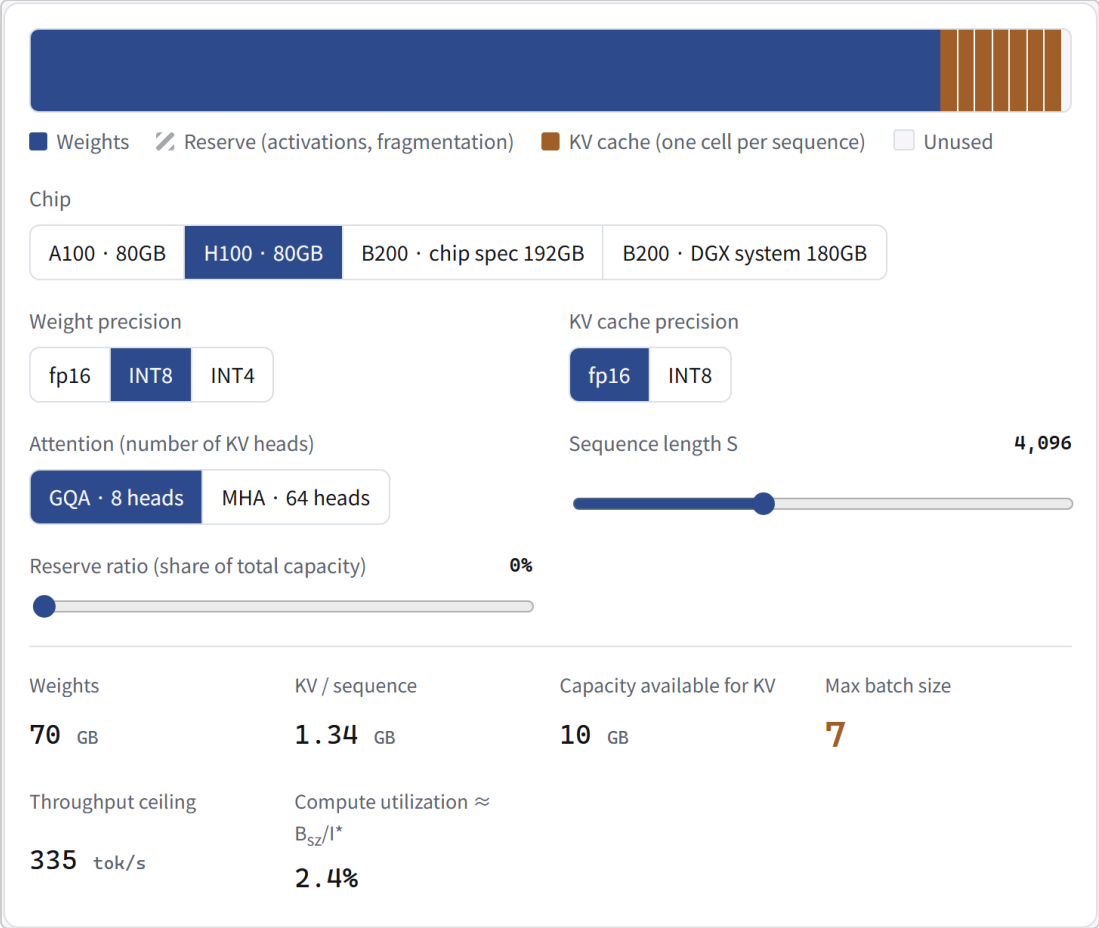


Figure 2-3. Load a 70B-class model onto one H100 80GB with INT8 weights (70 GB), take sequences of S = 4,096 with GQA and fp16 KV, and the KV cache has room for only 7 sequences. At batch size 7, assuming the math also runs in INT8, compute-unit utilization is about 2.4%. — In the HTML edition you can change the values here (HTML edition, Session 2, Solution to Exercise 3).

Solution to Exercise 4 — ASIC break-even

▾ Solution

$$N^* = \frac{C_{\text{NRE}}}{(c_G + o_G) - (c_A + o_A)}$$

Assumptions: NRE USD 100 million, ASIC unit price USD 3,000, GPU unit price USD 30,000. One ASIC replaces one GPU 1:1 (section 5). Electricity over 3 years is set up as follows: GPU 700W, ASIC 300W, PUE 1.3, 80% utilization (here, the share of time the chip draws its rated power), electricity at USD 0.08/kWh.

- GPU energy $= 0.7 \times 1.3 \times 8760 \times 3 \times 0.8 = 19{,}132$ kWh $\rightarrow o_G \approx$ USD 1,531
- ASIC energy $= 0.3 \times 1.3 \times 8760 \times 3 \times 0.8 = 8{,}199$ kWh $\rightarrow o_A \approx$ USD 656

$$N^* = \frac{10^8}{(30{,}000 + 1{,}531) - (3{,}000 + 656)} = \frac{10^8}{27{,}875} \approx 3{,}590 \text{ units}$$

What if the model's life is 2 years? Electricity costs fall to 2/3, the denominator

becomes 27,583 → N^* ≈ 3,630 units (3,625.4 by division). **It barely changes.**

Where to measure power — chip, server, facility. The 700 W above is the value the NVIDIA H100 spec sheet gives as the maximum TDP of the H100 SXM ("Up to 700W (configurable)"). It is the ceiling on the heat one chip can produce, not its average draw during work, and still less the power of the whole server. Power grows as the boundary widens.

BOUNDARY	VALUE	BASIS
Chip	Up to 700 W (H100 SXM TDP)	H100 spec sheet
Server	Up to 10.2 kW (DGX H100, 8 GPUs) → about 1.28 kW per GPU	Power specifications in the DGX H100/H200 User Guide. By calculation, $10.2 \div 8 = 1.275$ kW
Facility	Server power × PUE	This exercise's assumed PUE of 1.3

Counted again at the server boundary with the same assumptions (PUE 1.3, utilization 0.8, USD 0.08/kWh, 3 years):

- GPU energy $= 1.275 \times 1.3 \times 8760 \times 3 \times 0.8 \approx 34{,}847$ kWh → $o_G \approx$ USD 2,788 (about 1.8 times the chip boundary)
- For fairness the ASIC has to be raised to the same boundary. There is no basis for specifying an ASIC server, so raise it by the same ratio as the GPU ($1.275 / 0.7 \approx 1.82$ times) to about 0.546 kW ($0.3 \times 1.275 / 0.7 = 0.54643\ldots$). Then energy ≈ 14,935 kWh (computed without rounding the ratio) → $o_A \approx$ USD 1,195

$$N^* = \frac{10^8}{(30{,}000 + 2{,}788) - (3{,}000 + 1{,}195)} = \frac{10^8}{28{,}593} \approx 3{,}500 \text{ units}$$

If only the GPU is raised to the server boundary and the ASIC is left at its chip value of 300 W, the denominator becomes USD 29,132 and N^* ≈ 3,430 units, tilting toward the ASIC. Moving the boundary on one side only skews the comparison. Whichever way you count, N^* stays at 3,400–3,600 units, and at the server boundary the electricity difference of USD 1,593 is about 6% of the denominator, so the conclusion (acquisition cost dominates) stands. One more example at the same boundary: for the Blackwell generation, the DGX B200 datasheet gives the maximum power of the 8-GPU system as about 14.3 kW (about 1.79 kW per GPU by calculation). Unit prices, electricity rates, PUE and utilization are still illustrative assumed values; only the power ceilings of the chip and the server come from manufacturer specifications.

What this exercise really teaches — the result runs against intuition

1. **The break-even volume is surprisingly small.** 3,600 units is a few weeks' deployment for a hyperscaler. On pure accounting, an in-house chip wins by a wide margin.
2. **Electricity is not decisive.** Of the USD 27,875 denominator, the electricity difference is USD 875, a mere 3% (about 6% even at the server boundary). What dominates is **the USD 27,000 difference in acquisition cost.** The common explanation "they build their own chips to save on electricity" does not fit these numbers.
3. **Then why do most not do it?** Because N^* is not the answer. The real constraint is **time.** Development takes 2–3 years, and at completion the computation pattern the chip was aimed at must still dominate. The probability that a design decision made 2 years ago gets today's model structures right (MoE, long contexts, new attention variants) is the real decision variable.

So the hyperscalers' real advantage lies not in volume but in "controlling their own workloads." If, instead of fitting the chip to the model, **they can fit the model to the chip**, the part of the time risk above that comes from "the design failing to anticipate the model structure" shrinks a great deal. Risks such as whether demand arrives as expected, or where the performance that competing models require will move, remain. The software lock-in discussion of section 7 is completed here.

> In section 5 of Session 4, c_A is computed directly from the yield equation to refine this formula.

The figure below counts the assumed values of Exercise 4 (NRE USD 100 million, ASIC USD 3,000, GPU USD 30,000, PUE 1.3, 80% utilization, USD 0.08/kWh, 3 years) at the chip boundary (GPU 700 W, ASIC 300 W). In the screen edition you can switch the power boundary among chip, server (both sides) and server (GPU only), and move the unit prices, power, PUE, utilization, electricity rate and period to see how N^*, the crossing point of the two total-cost lines, and the electricity share of the denominator change.

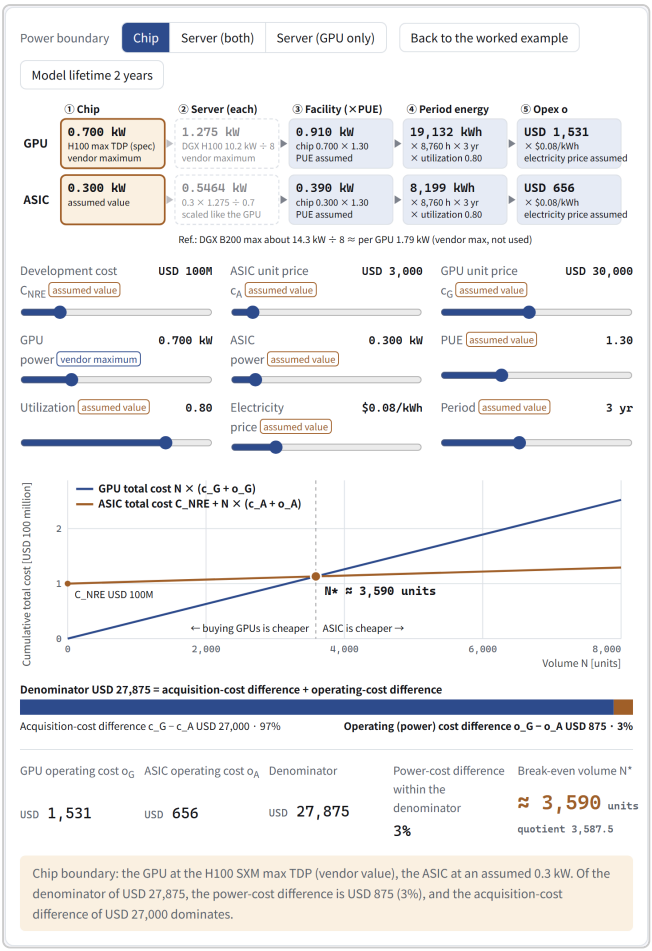


Figure 2-4. With the chip boundary (GPU 700 W, ASIC 300 W) and the assumptions of Exercise 4 (NRE USD 100 million, unit prices USD 3,000 and 30,000, PUE 1.3, 80% utilization, USD 0.08/kWh, 3 years), the two total-cost lines meet at N* ≈ 3,590 units. Of the USD 27,875 denominator, the power-cost difference is USD 875 (3%). — In the HTML edition you can change the values here (HTML edition, Session 2, Solution to Exercise 4).

Self-study appendix B — leads for the discussion questions

QUESTION	KEY AXIS	LEADS
1. Limits of the roofline	**The steady-state assumption**	The roofline is the ceiling for "a kernel that runs long enough." It captures none of the following: latency (time to the first token), the cache hierarchy (flattened into a single B), cross-node communication, kernel launch and synchronization overhead, and clock throttling from power limits. The decoding example in section 6 assumed that only the weights are read, so the traffic from reading the KV cache is also missing, and with a long context the real arithmetic intensity is lower still (Exercise 3). The peaks of low-precision arithmetic units were also simplified. The roofline figure of section 6 keeps P_{peak} at the BF16 dense value even for INT8 and INT4 weights; if the computation runs at the same precision, the peak and the arithmetic intensity grow together and utilization ends up similar to fp16 (solution to Exercise 3). For the size of launch overhead, the example NVIDIA measured in its 2019 CUDA Graphs post on a V100 with CUDA 10.1 gives a feel. Launching a short kernel that runs in 2.9 µs and synchronizing after each one took 9.6 µs per kernel; cutting synchronization to once per step brought it to 3.8 µs; launching 20 kernels bundled into one CUDA Graph brought it to 3.4 µs. These are not figures for current GPUs but an example of scale. When kernels get as short as microseconds, the time to launch and wait can exceed the run time. **It is a model that gives a ceiling, not a predictive model**
2. Is $I^* \approx 300$ a coincidence?	**Design intent**	No coincidence. If bandwidth cannot be raised to keep up, the chip ends up with idle compute units, so vendors keep the balance. Move above it (too much compute) and real-world efficiency falls; move below it (too much bandwidth) and HBM costs eat into profit
3. MoE on the roofline	**Weights read vs. weights held**	Total parameters grow, so **capacity** pressure rises; active parameters shrink, so the **decoding ceiling** rises. It is a structure that trades a capacity constraint for a bandwidth constraint. But routing creates irregular access that lowers effective bandwidth utilization, and when experts scatter within a batch the gain is halved
4. How long software lock-in lasts	**Who writes the kernels**	What lock-in really consists of is not the language but **the accumulation of optimized kernels.** If Triton and XLA lower the cost of writing kernels, that accumulation is worth less. Conversely, the more specialized the hardware becomes (dedicated instructions, special memory hierarchies), the harder it is for compilers to absorb, and lock-in grows stronger
5. Designing an inference-only chip	**Where to put the ridge point**	Decoding is low at $I \approx B_{\text{sz}}$, so it is rational to cut compute, put the budget into bandwidth and capacity, and design I^* **low.** Fix precision mainly at INT8/INT4 and spend the area that frees up on on-chip SRAM. This is the direction actual inference-specialized chips have taken

Self-study appendix C — key terms

TERM	ONE-LINE DEFINITION
Roofline model	A model that expresses the performance ceiling as the minimum of the compute limit and the bandwidth limit
Ridge point	The arithmetic intensity $I^* = P_{\text{peak}} / B$ at which compute-bound and memory-bound divide

TERM	ONE-LINE DEFINITION
Compute-bound	A state in which the compute units are the limit ($I > I^*$)
Memory-bound	A state in which memory transfer is the limit ($I < I^*$)
GEMM (general matrix multiply)	General matrix-matrix multiplication
Prefill / decode	The phase that processes the whole input at once / the phase that generates tokens one at a time
Batch size	The number of sequences processed at once
Tiling / blocking	An optimization that cuts data into pieces small enough to load on chip and reuses them
Kernel fusion	An optimization that merges several operations so that intermediate results are not written to memory
FlashAttention	The leading kernel that fuses attention to cut round trips to HBM
MoE (mixture of experts)	A sparse model structure that activates only some experts for each token
GQA / MQA (grouped / multi-query attention)	Attention variants that reduce the number of KV heads to shrink the KV cache
Systolic array	A compute structure that streams data through a grid and repeats multiply-accumulate operations
Structured sparsity	A technique that sets some weights to 0 to cut computation. The main culprit in inflated spec-sheet figures
NRE (non-recurring engineering)	The one-time cost of developing a chip
Tensor parallelism	A scheme that splits the matrices of one layer across several accelerators
TTFT (time to first token)	The time from the request to the first token. Governed by prefill
TPOT (time per output token)	The average interval between output tokens. Governed by decoding
P99 (99th percentile latency)	The latency below which 99% of requests finish. A measure of tail latency
Fallback	Handing an operation the accelerator does not support to the CPU to run. A device-host round trip is added

---

Self-study appendix D — self-assessment

☐ I can write $P_{\text{attainable}} = \min(P_{\text{peak}}, B \times I)$ without looking and say what each term is → section 6 ★

☐ Given any chip's spec sheet, I can compute I^* within 30 seconds → Exercise 1 ★

☐ I can explain **why** prefill is compute-bound and decoding memory-bound, using the I of each → section 6

☐ Using the decoding-ceiling formula, I can give two reasons why "70B fp16 on one H100" is impossible → Exercise 2

☐ I can explain why quantization improves **both** the ceiling and capacity → Exercise 2

☐ I can set up the KV cache size formula and explain how capacity limits bandwidth efficiency → Exercise 3 ★

☐ I can tell ways of raising arithmetic intensity from ways of raising bandwidth and name three of each → the last table in section 6

☐ I can show by calculation that in ASIC break-even **acquisition cost, not electricity, dominates** → Exercise 4

☐ I can name three things the roofline **cannot** explain → appendix B

If you do not pass the three ★ items, do not move on to Session 3. All of Session 3 is the answer to "so how do we raise *B*?"

Before the next session

The conclusion of this session fits in one sentence. **For work whose arithmetic intensity falls short of the ridge point, small-batch decoding above all, performance is set by bandwidth, not by compute.** Work with high arithmetic intensity, such as prefill and training, is still compute-bound.

That leads naturally to the next questions.

QUESTION THIS SESSION LEAVES	THE ANSWER IN SESSION 3
What does it take to raise *B*?	Widen the bus or raise the pin speed, and why the former was chosen
Why is that so expensive?	Stacking, TSVs, KGD screening, bonding, and the cruel arithmetic of stack yield
How is the capacity constraint (Exercise 3) solved?	HBF and memory tiering

---

Sources

Introductory material chosen by the study curriculum (the "AI chips (compute semiconductors)" item)

CURRICULUM SUBTOPIC	LINK	HOW THIS BOOK USES IT
What an AI chip is	Element Korea — "What Is an AI Chip? Why the World Is Watching, and the Difference Between NPUs and GPUs" (in Korean)	Introductory overview of the concepts. An easy explanation matching the CPU/GPU/NPU distinctions of sections 2 and 5. This book goes further and recasts them from the angle of energy and overhead

CURRICULUM SUBTOPIC	LINK	HOW THIS BOOK USES IT
What an AI chip is	MSAP — "What Is an AI Accelerator? GPU, NPU or TPU: Which Will Prevail?" (in Korean)	Overview of the types of AI accelerator. Background to the flexibility-efficiency spectrum of section 5
What an AI chip is	ScienceON JAKO202412457639314 — "Research Trends in Domestic and International AI chips" (in Korean)	Korean academic literature. Overview of AI semiconductor technology
The AI chip market	Tilnote — "NVIDIA GPUs vs. Google and Amazon AI Chips: What Is the Difference?" (in Korean)	Summary of market trends. Recommended as material for the "reading numbers critically" exercise in section 4 of Session 1
The AI chip market	ScienceON JAKO202504350488400 — "A Comprehensive Study of AI Semiconductors and Key Application Fields" (in Korean)	Korean academic literature. Analysis of the AI semiconductor market
The AI chip market	InitNode (an article first published on eigenstate.dev and later moved) — "CPU vs GPU vs TPU vs NPU: AI Hardware Architecture Guide 2026"	An in-depth English explainer comparing the four architectures. **The material that maps most directly onto the structural comparisons of sections 3 and 4, recommended both before and after the session**

Verified (2026-08-02)

- Horowitz (ISSCC 2014) energy figures — widely cited standard reference values (which values appear in the original talk is covered in the 2026-10-08 item below). **Because they are for a 45nm process, use them to grasp the size of the ratio (two orders of magnitude), not as absolute values**
- NVIDIA H100 SXM: BF16 dense 989 TFLOP/s, HBM3 80 GB, 3.35 TB/s — manufacturer's published specifications
- NVIDIA B200: BF16 dense 2,250 TFLOP/s, HBM3E 192 GB, 7.7 TB/s — manufacturer's published specifications (Blackwell Architecture Technical Brief V2.1, Table 3, HGX B200). **192 GB and 7.7 TB/s are the chip's published (maximum) values.** The DGX B200 system has 180 GB and 8.0 TB/s per GPU (64 TB/s total ÷ 8). The system specifications differ — see the 2026-10-08 item below
- TPU v1 256 × 256 MAC array — Jouppi et al., ISCA 2017, original paper

Verified (2026-10-08) — all of the following were checked by opening the primary sources directly.

- Chip vs. system distinction for B200 memory capacity — "Up to 192 GB HBM3e" per HGX B200 GPU in the NVIDIA Blackwell Architecture Technical Brief V2.1; 1,440 GB total for 8 GPUs (180 GB per GPU by calculation) in the DGX B200 User Guide and the DGX B200 datasheet (English edition: the Technical Brief wording rechecked against the PDF, 2026-10-22)
- Power boundaries — H100 SXM maximum TDP 700 W (H100 spec sheet), DGX H100 maximum 10.2 kW (DGX H100/H200 User Guide), DGX B200 maximum about 14.3 kW (DGX B200 datasheet and user guide). The server-boundary recalculation in Exercise 4 was redone with these values and the assumed values of Exercise 4

- H100 SXM PCIe Gen5 128 GB/s and GPU memory 3.35 TB/s — official figures on the H100 spec sheet. The H100 white paper V1.04 states that 128 GB/s is the total for both directions (64 GB/s each way)
- Fallback behavior — the description of the priority list in the ONNX Runtime execution providers documentation, and `PYTORCH_ENABLE_MPS_FALLBACK` in the PyTorch 2.14 MPS environment variables documentation
- The structure of `torch.compile` (TorchDynamo, TorchInductor, Triton) — PyTorch 2.14 torch.compiler documentation
- Kernel launch overhead of 2.9, 9.6, 3.8 and 3.4 µs — the text of the NVIDIA technical blog post "Getting Started with CUDA Graphs" (2019-09-05). Tesla V100, CUDA 10.1
- Intel AMX tile configuration (8 tiles, 8 KB in total, at most 16 rows × 64 bytes) and TMUL — Intel AMX code-sample article
- The accuracy floor (99.9% of the FP32 reference) and TTFT/TPOT constraints of the MLPerf Inference Llama2-70B task, the definitions of TTFT and TPOT, and the 99th percentile in the Server and Interactive scenarios — MLCommons inference rules, master version (last modified 2026-08-20, commit ff7edba)
- DeepSeek-V3's 2,048 H800s, 2.788M GPU hours (2,664K + 119K + 5K), USD 5.576 million at an assumed USD 2 per hour, the caveat excluding prior research and ablation experiments, DualPipe, and 20 of 132 SMs — text of arXiv:2412.19437 v2
- Systolic-array beat counts and utilization at TPU v1 size (0.2%, about 89%) — recalculated with the same formula as the interactive figure, $M\ /\ (M + K + N - 2)$
- Energy table in section 1 — checked in a public copy of Horowitz's paper (ISSCC 2014, paper 1.1), Figure 1.1.9, "Rough energy costs for various operations in 45nm 0.9V": 8-bit integer add 0.03 pJ, 32-bit floating-point add 0.9 pJ and multiply 3.7 pJ, cache (64-bit access) 8 KB 10 pJ, 32 KB 20 pJ, 1 MB 100 pJ, DRAM 1.3–2.6 nJ. The 32-bit SRAM 5 pJ and DRAM 640 pJ are not in this figure; they are in Figure 1 ("Energy table for 45nm CMOS process", citing Horowitz) of Han et al. (2015), arXiv:1506.02626 v3
- Memory coalescing with 32-byte transactions, 4 consecutive bytes × 32 threads = 128 bytes = 4 transactions, 32 at worst — CUDA Programming Guide §2.3.4.1. §2.3.3.2 and §2.3.3.6 of the same chapter describe the L1 and L2 caches and shared memory
- H100 SXM INT8 tensor core 3,958 TOPS (with sparsity; dense is half that, 1,979, 2 times the BF16 dense 989) — H100 spec sheet. The spec sheet gives no INT4 value (the INT4 peak in Exercise 3 is an assumption)

Unconfirmed · reserved

- Specific amounts for ASIC NRE and mask-set costs — they vary widely by node, company and contract and are often not disclosed. The text mentions them only by order of magnitude, and the exercise states that they are **assumed values**
- AI accelerator shares by company — not asserted, as in Session 1
- Unit prices and electricity costs in the section 5 exercise — all illustrative assumed values, not real market prices

SESSION 3

AI Memory — DRAM · HBM · HBF

KEY TAKEAWAY

HBM's odd-looking shape (vertical stacking + a 2,048-bit ultra-wide bus + a silicon interposer) is not a design choice. It follows inevitably from physical constraints. There are only two ways to raise bandwidth: "send each signal faster" or "add more wires." The first ran into walls of signal integrity and power. So the industry pushed the second to the extreme, and pulling out thousands of wires means placing the parts physically very close together, which made TSVs and interposers indispensable. **That is why the memory problem became a packaging problem.** HBM is expensive because yields multiply (a 12-high stack = die yield to the 12th power). HBF appeared because, apart from bandwidth, **capacity** is a bottleneck of its own.

PRIMARY SOURCES

Semiconductor Network, "The Evolution of Memory Semiconductors in the AI Era… from Flash, SRAM and DRAM to HBM" (in Korean) · Rambus, "High Bandwidth Memory (HBM): Everything You Need to Know" · HyperAccel, "Understanding HBF" (in Korean) · ETRI, *Electronics and Telecommunications Trends* vol. 41 no. 3, "Key Technologies and Market Trends in Artificial Intelligence (AI) Memory Semiconductors" (in Korean) · JEDEC JESD270-4 (HBM4) press release · Samsung Electronics press release on shipping HBM4 in volume (2026-02-12) · SK hynix, first HBF standard specifications (2026-08-04) · Micron 16Gb DDR4 data sheet Rev. H

Learning objectives

1. Explain why the different device structures of SRAM, DRAM and NAND produce their characteristic speed, capacity and cost.
2. Explain why DRAM latency has barely improved in 20 years.
3. Reconstruct the structure of HBM from the question "why does it look like this?"
4. Calculate the multiplicative structure of stack yield and explain the cost structure of HBM.
5. Distinguish the bottleneck HBF targets from the one HBM targets.

0. Self-study guide — for readers working alone

> **THIS SESSION IS THE ANSWER TO SESSION 2**
>
> Session 2 ended with "in small-batch decoding, bandwidth sets performance." Then why not just raise the bandwidth? This session shows why that "just raise it" is a problem that shakes the whole industry. **Work through the stack-yield arithmetic (Exercise 1) and the price of HBM shows up in numbers.**

What carries over from the previous session

FROM SESSION 2	IN THIS SESSION
$I^* = P_{\text{peak}} / B$	How B is physically built
"Decoding is memory-bound"	Hence the need for HBM
The capacity constraint in Exercise 3	Hence the discussion of HBF (high capacity)

Prerequisite check

CONCEPT	ONE-LINE ANSWER
DRAM cell	Stores a 0 or 1 as charge in a single capacitor. The charge leaks away, so it is rewritten periodically (refresh)
NAND flash	Traps charge inside an insulating layer, so data survives power-off. In exchange it is slow and has a limited write life
Bus width / transfer rate	How many lanes carry data at once / how many times per second one lane sends. **Bandwidth = the product of the two**
TSV	Through-silicon via. An electrical path that runs vertically through the silicon
pJ/bit	The energy needed to move 1 bit. The real currency of the bandwidth era

Reading order (110 minutes in total recommended)

STEP	SECTION	TIME	NOTE
1	Memory hierarchy, DRAM basics	25 min	Background
2	**Two ways to build bandwidth**	20 min	Much faster if read together with Exercise 2
3	**HBM structure and stack yield**	30 min	**Read closely.** Do Exercise 1 right here
4	Section 6, HBF	20 min	Capacity, a different axis
5	Section 7, PIM/PNM + appendices	15 min	

The one sentence that runs through this session

> **Bandwidth and capacity are different constraints, and different technologies solve them.** Bandwidth: HBM (close and wide). Capacity: HBF and tiering (farther away but large). Mix the two up and this session will confuse you throughout.

A common sticking point — why "wide and slow"?

Intuitively, raising pin speed looks easier. Isn't making one lane twice as fast simpler than planting 1,024 pins?

No. Raising the transfer rate makes **energy per bit grow nonlinearly** (equalization circuits, clock recovery, driver strength), and signal-integrity problems get sharply worse. Build 4 TB/s of bandwidth from 1,024 pins at 32 Gbps and the interface alone can eat close to 100W per stack (96W at 3 pJ/bit). With 8 stacks on one accelerator, the memory interface alone takes about 770W, a second heat source on par with the GPU's own power (700–1,000W). How much the temperature actually rises depends on the area over which that heat spreads and on the cooling, but it is hard to give the memory side a power budget of that size. So HBM chose to use **very many slow pins**, and the price was that it needed room to plant them, which brought in **stacking, TSVs and interposers**. Exercise 2 checks this logic with numbers.

---

1. Getting the intuition — the memory hierarchy is a map of physical compromises

Readers who have studied computer architecture will know the hierarchy "registers → L1 → L2 → L3 → DRAM → SSD." Such courses usually do not cover **why** the hierarchy splits into those sizes and speeds. The answer lies in the device structure.

LEVEL	DEVICE STRUCTURE	TRANSISTORS PER CELL	RESULT
SRAM (cache)	Cross-coupled inverter latch	6 (6T)	Fast, not nonvolatile, **large area → small capacity and expensive**
DRAM (main memory)	1 transistor + 1 capacitor	1 (1T1C)	Small cell, so **dense and cheap**; in exchange slower and needs refresh
NAND (storage)	Transistor with a charge-storage layer, stacked vertically	1, but 3–4 bits per cell (TLC, QLC)	**Ultra-dense and nonvolatile**; in exchange erased block by block, slow, finite endurance

One-sentence summary: the number of devices it takes to hold one bit sets the character of the whole hierarchy. As transistors per bit fall from 6 for SRAM → 1 for DRAM (plus one capacitor) → 1/3–1/4 for NAND (3–4 bits per cell), memory gets cheaper and denser but slower and harder to handle. NAND then stacks cells vertically in hundreds of layers, multiplying the bits held in the same area once more. Cutting the device count and stacking layers are separate sources of density.

The hierarchy as seen from an AI accelerator (as of October 2026)

Inside an AI server the same devices sit in the order below. The higher the level, the shorter the latency; the lower, the larger and cheaper. Looking at stack bandwidth alone, HBF can exceed host DRAM, but NAND's long read latency puts it a level below DRAM. In its keynote at FMS in August 2026, SK hynix presented the memory hierarchy as four levels: HBM, DRAM, NAND (HBF) and SSD. This table adds on-chip SRAM on top of that order.

LEVEL	MEDIUM	WHERE IT SITS	WHAT IT DOES
On-chip SRAM	SRAM	Inside the accelerator die	The tile currently being computed
HBM	Stacked DRAM	Right beside the accelerator (on the interposer)	Weights and KV cache read for every token
DRAM	DDR, LPDDR (including CXL expansion memory)	Memory slots on the CPU side, expansion devices	Host memory, overflow KV cache
HBF	Stacked NAND	Connected to the processor over UCIe (section 6)	Large data that rarely changes (proposal stage)
SSD	NAND	The server's storage devices	Model files, datasets

The order is by speed and capacity, not by physical distance. HBF sits below DRAM, yet it is meant to be placed close to the accelerator as HBM is. As an analogy, SRAM is the top of your desk, HBM the bookshelf within arm's reach, DRAM the stacks in the same room, and SSD the basement archive. HBF is a large new set of shelves placed next to the desk: close by, but slower to pull a book from than the stacks. The analogy captures only distance, retrieval speed and size. NAND's wear with every write and the fact that it can be erased only block by block have no counterpart in the book analogy.

---

2. The physics of DRAM — why it is slow and why it does not get faster

What a read actually does

A DRAM cell stores 1 bit as the presence or absence of charge on a very small capacitor. To read it:

1. **Wordline activation** — turn on the access transistors of the row
2. **Charge sharing** — the charge on the cell capacitor flows out onto the bitline and shifts the bitline voltage very slightly. Cell capacitance is far smaller than bitline capacitance, so the change is only on the order of tens of mV
3. **Sense amplification** — compare against a reference voltage and amplify that tiny difference to full swing
4. **Reading = destroying** — the charge has left, so the original value **must be written back** (restore)
5. **Precharge** — return the bitline to the midpoint voltage for the next access

What the timing parameters mean

This procedure is exactly what DRAM timing is made of.

PARAMETER	PHYSICAL PROCESS IT CORRESPONDS TO
tRCD (RAS to CAS delay)	Wordline activation → sense amplification complete
tCL (CAS latency)	Column address → data out
tRP (row precharge)	Time to complete precharge
tRAS (row active time)	Minimum time a row stays open, including restore
tREFI / tRFC	Refresh interval / time a refresh takes

The key fact: these are analog physical processes governed by charge and capacitance, so process scaling does not speed them up much. If anything, smaller cells store less charge and sensing gets harder.

> **Twenty years of asymmetry:** DRAM **bandwidth** has grown dozens of times over, but **latency** has stayed nearly the same. Bandwidth can be raised through parallelism and interface speed, but latency is set by the physical process in a single cell. This asymmetry pushed all of computer architecture toward "hiding latency" (prefetching, multiple banks, the massive thread parallelism of GPUs).

The refresh tax

Charge leaks, so cells must be read and rewritten periodically. During refresh the bank cannot be accessed, which eats into bandwidth and power. The burden grows with capacity, making it a standing problem in high-capacity DRAM design.

It is easy to confuse this with step 4 above (restore after a read). Restore is caused by **an access**; refresh is caused by **time**. Even a cell nobody reads must be refilled once within a fixed period.

Worked example — how large is the tax? Take the Micron 16Gb DDR4 data sheet (Rev. H, 2021). At or below 85°C, 8,192 REFRESH commands must be issued within 64ms, so the command interval tREFI is 7.8μs (64ms ÷ 8,192 = 7.8125μs). Between 85 and 95°C it is 8,192 within 32ms, so tREFI halves to 3.9μs. The time one command ties up the chip, tRFC1, depends on density. The fraction of time the chip is tied up in refresh is roughly tRFC/tREFI.

DIE DENSITY	TRFC1	AT OR BELOW 85°C (TREFI 7.8MS)	85–95°C (TREFI 3.9MS)
2Gb	160 ns	2.0%	4.1%
4Gb	260 ns	3.3%	6.7%
8Gb	350 ns	4.5%	9.0%

While density quadrupled, the tax more than doubled, and it doubles again when the chip runs hot. These ratios divide tRFC1 by 7.8125μs and 3.90625μs. It is a simple calculation that assumes the whole chip can do nothing else during refresh.

3. Two ways to raise bandwidth, and the dead end

Memory bandwidth is approximately

$$B = (\text{bus width}) \times (\text{transfer rate per pin})$$

So there are only two methods.

Think of a road: bus width is the number of lanes, and the per-pin transfer rate is how fast the cars go. To get more cars past per hour, you add lanes or make the cars go faster. The analogy misses something. On a road, faster cars do not make sending one car steeply more expensive, but a memory pin spends steeply more energy per bit (pJ/bit) the faster it sends. That cost is why the two paths below part ways.

Path ①: raise the per-pin transfer rate (the GDDR path)

- Toggle the signal faster. GDDR6X and GDDR7 go as far as multilevel signaling (PAM3/PAM4).
- How fast has it become? The JEDEC GDDR7 standard (JESD239, published 2024-03-05) uses PAM3, which sends 3 bits in 2 cycles using three voltage levels, and reaches up to 192 GB/s per device. For a 32-bit device that works out to 48 Gbps per pin, 6 times the HBM4 standard pin speed of 8 Gbps. HBM does not use slow pins because pin speed cannot be raised.
- **The limit:** raising the frequency makes signal integrity deteriorate sharply. Loss, reflection and crosstalk in board traces all worsen with frequency, and the equalization and serialization circuits that compensate eat power. **The decisive problem is that energy per transferred bit (pJ/bit) goes up.**
- At data-center scale that power is both operating cost and heat, so this path alone cannot get there.

Path ②: widen the bus (the HBM path)

- A DDR5 channel is 64 bits and GDDR is 32 bits per device, while **an HBM stack is 1,024 bits (through HBM3), and HBM4 is 2,048 bits.**
- The same bandwidth can be had at a much lower frequency, so **pJ/bit falls substantially.**
- **The limit:** to pull out thousands of wires, the wires need somewhere to go. An ordinary PCB substrate cannot reach that wiring density, and a package does not have enough pins either.
- **A caveat — there is also a standard that mixes the two paths.** SPHBM4 (JESD330-4), published by JEDEC on July 13, 2026, uses the same DRAM dies as HBM4 but changes the base die, cutting the data signals from 2,048 to 512 and using 4:1 serialization to reach the same aggregate bandwidth. With fewer pins the bump pitch can be relaxed, so the stack can be mounted on an ordinary organic substrate instead of a silicon interposer, and the memory can sit farther from the SoC. "A wide bus requires an interposer" holds when all

2,048 signals are used; pay the price of running each signal four times faster and you can move partway back toward path ①. What that price is in pJ/bit is not in the press release.

The answer that emerged

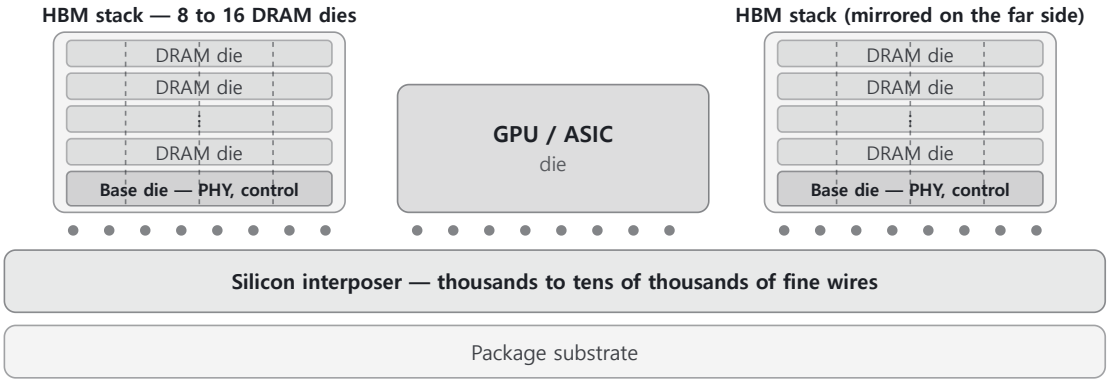
1. To pull out a very large number of wires → you need a medium that allows extremely dense wiring → **a silicon interposer**
2. To connect several dies inside a stack → **TSVs (through-silicon vias)**
3. Shorter distance means less power and latency → **attach it right beside the GPU (2.5D packaging)**

At that moment the memory problem became a packaging problem. That is why Session 6 is the most important session in this book.

---

4. HBM structure in detail

Physical layout



Vertical dashed lines = TSVs (through-silicon vias) · dots = microbumps / hybrid bonding

What each element does

- **Stacked DRAM dies:** capacity comes from height rather than area. 8-high, 12-high, 16-high.
- **TSVs:** vertical wiring that runs through each die. They bind the whole stack into one wide bus.
- **Base die:** a layer made in a logic process, not a DRAM process. It holds the PHY (physical-layer interface), test circuits and control logic. **With HBM4, making this base die in a foundry logic process has become reality.** Samsung Electronics used a 1c DRAM core and a 4nm logic process in the HBM4 it began shipping in volume in February 2026, and its May 2026 press release on HBM4E samples stated that the base die is "Samsung Foundry's 4nm logic base die" and that HBM4E uses "the same combination of core and base die" as HBM4. This is where the line between memory makers and foundries blurs.
- **Die-to-die bonding — microbumps and what comes next:** HBM in volume production

today joins dies with microbumps and fills the gaps between them. SK hynix used MR-MUF (pouring a liquid protective material between the stacked chips and curing it) for HBM4 (announced 2025-09), and Samsung Electronics used TC-NCF (laying a non-conductive film on each layer and thermocompression bonding it) for its 12-high HBM3E (announced 2024-02). Hybrid bonding, which joins copper to copper directly without bumps, is under development as the next candidate for taller stacks; as of October 2026 we found no manufacturer announcement of volume production of HBM using it. Hybrid bonding itself is covered in Session 6.

- **Interposer:** the silicon plate that actually draws the thousands of wires between the GPU and the HBM.

Specifications by generation

GENERATION	INTERFACE WIDTH	CHANNELS	STACK BANDWIDTH — PER STANDARD	STACK BANDWIDTH — VOLUME PRODUCT (ANNOUNCEMENT DATE)
HBM2E	1,024 bits	8	~0.4–0.5 TB/s (approximate)	
HBM3	1,024 bits	16	~0.8 TB/s (approximate)	
HBM3E	1,024 bits	16		Pin up to 9.6 Gbps → about 1.2 TB/s by calculation (comparison value in Samsung's HBM4 press release, 2026-02-12). Samsung's HBM3E 12H press release (2024-02-27) gives up to 1,280 GB/s
HBM4	**2,048 bits**	**32** (2 pseudo-channels per channel)	**8 Gbps → up to about 2 TB/s** (JESD270-4, 2025-04-16)	Pin 11.7 Gbps (up to 13 Gbps), up to 3.3 TB/s per stack (Samsung, volume shipment 2026-02-12)
HBM4E				Pin 14 Gbps (up to 16 Gbps), up to 3.6 TB/s per stack, 12-high 48 GB (Samsung, sample shipment 2026-05-29)

The standard column gives what the pin speed set by the JEDEC standard can deliver; the product column gives values the manufacturers announced. As of October 2026, volume HBM4 exceeds the standard pin speed.

- **SK hynix** announced on September 12, 2025, that it had completed HBM4 development and readied mass production: 2,048 I/O, over 10 Gbps per pin (above the JEDEC standard of 8 Gbps), 1b DRAM (the fifth generation of the 10nm class), and the Advanced MR-MUF process.
- **Samsung Electronics** announced volume shipments of HBM4 on February 12, 2026. It runs at 11.7 Gbps per pin, can be pushed to 13 Gbps, and comes as a 12-high stack of 24–36 GB. Multiply 11.7 Gbps by 2,048 bits and you get about 3.0 TB/s; at 13 Gbps, about 3.3 TB/s. The press release's "maximum of 3.3 terabytes-per-second (TB/s)" per stack is the 13 Gbps figure.
- The same company's 12-high HBM4E samples (2026-05-29): assume 2,048 bits as for HBM4 and multiply by 14 Gbps, and you get about 3.6 TB/s, matching the press release (which

does not state the interface width).
- The standard-column values for HBM2E and HBM3 and the HBM3E row are rough ranges, because pin speed differs by vendor and product. Blank cells were not confirmed from a primary source for this edition.

Main changes in HBM4

- Interface width 1,024 → **2,048 bits** (2 times)
- Independent channels 16 → **32**, each with 2 pseudo-channels
- Backward compatibility with HBM3 controllers — the interface is defined so that a single controller can handle both HBM3 and HBM4
- DRFM (directed refresh management) introduced — row-hammer mitigation and better reliability (RAS)
- Supports 4- to 16-high stacks, die density 24Gb or 32Gb → 32Gb at 16-high gives 64 GB per cube
- Lower core voltage (1.1V in HBM3/3E → 1.0/1.05V); VDDQ chosen by the vendor from 0.7–0.9V

> **Note:** HBM4 **doubled the width** instead of raising the frequency sharply. That means it chose "path ②" from section 3 once again. Power efficiency is why this path keeps being chosen.

Why each GPU carries several HBM stacks

Because one stack is short on both bandwidth and capacity. The latest accelerators arrange 6–8 HBM stacks around the GPU die. As one example, NVIDIA said the Rubin GPU, with 12-high HBM4 stacks, delivers up to 288 GB and up to 22 TB/s per GPU (technical blog, 2026-07-21; 2.8 times Blackwell's bandwidth). The post does not give the number of stacks. Assuming 36 GB stacks, there are 8, which gives 2.75 TB/s per stack and about 10.7 Gbps per pin — an estimate worked backward, nothing more. But adding stacks:

- enlarges the interposer → the interposer, too, is limited by the lithography exposure field → extension methods such as CoWoS-L are needed
- makes the package bigger and raises heat density
- **makes packaging capacity the ceiling on AI chip shipments** (what the supply-chain bottleneck in Session 1 actually is)

---

5. Why HBM is expensive — yields multiply

For readers trained in statistics, this is the most intuitive section.

The basic calculation

Let y be the yield of one DRAM die, and stack n of them. If dies are simply picked at random and stacked, the probability that the whole stack works is

$$Y_{\text{stack}} = y^n \times \prod_{i=1}^{n-1} y_{\text{bond},\,i}$$

where y_{bond} is the yield of each bonding step.

Numerical example ($y = 0.99$, bonding yield 0.995)

STACK HEIGHT	DIE TERM y^n	BONDING TERM	STACK YIELD
4-high	0.961	0.985	0.946
8-high	0.923	0.966	0.891
12-high	0.886	0.946	0.839
16-high	0.851	0.928	0.790

The stack-yield column multiplies the unrounded values, so multiplying the two printed columns may differ in the last digit.

Even with an excellent 99% per die, a 16-high stack throws away 1 stack in 5. And when a stack is thrown away, its good dies go with it. If one die is bad, the other 15 good dies are discarded; if all 16 dies are good but one bond fails, all 16 are discarded. This is the heart of HBM's price structure.

Countermeasures

1. **KGD (known good die) screening** — test each die fully before stacking and use only good ones. This pushes y in the formula above toward 1. But a full test at wafer level needs separate test structures and time, and that is itself a cost.
2. **Redundancy** — provide spare rows, columns and TSVs to replace defects. DRAM has traditionally been good at repair.
3. **Better bonding yield** — one of the reasons hybrid bonding is being considered as the next candidate.

The cost structure in brief

HBM costs several times as much as ordinary DRAM of the same capacity because of a combination of (a) low stack yield, (b) extra steps such as TSV formation, thinning and bonding, (c) the cost of the interposer and advanced packaging, and (d) a market in which supply is limited to three companies.

> **For discussion:** which of these terms improves over time, and which remains structurally?

---

6. HBF (high bandwidth flash) — capacity, another bottleneck

Why it appeared

Look again at the decoding ceiling from Session 2.

$$\text{tokens/s} \lesssim \frac{B}{P \cdot b} \times B_{sz}$$

Raising the batch size B_{sz} raises throughput, but a larger batch needs **memory capacity to hold the KV cache.** And as the model itself grows, it needs capacity to hold the weights. **Bandwidth and capacity are separate constraints.**

HBM is optimized for bandwidth, but its capacity is expensive (around 24–32Gb per die). NAND, by contrast, is overwhelmingly denser. **"What if we stacked NAND like HBM and gave it a wide bus?"** is the idea behind HBF.

Status and specifications

- On August 6, 2025, **Sandisk and SK hynix announced a collaboration (memorandum of understanding) to standardize HBF.**
- On February 25, 2026, the two companies began standardization work as **a dedicated workstream within the OCP (Open Compute Project).**
- On August 4, 2026, the two companies **released the first HBF standard specification through the OCP** (SK hynix announcement). Two configurations, 8-high and 16-high, give up to 512 GB per stack; bandwidth comes in three grades (Grade 1–3) of about 0.4–3.0 TB/s; the connection to the processor is UCIe, the chiplet interconnect standard. Google and Tenstorrent have joined the consortium. The value for the middle grade could not be confirmed from public material.
- Sandisk's earlier first-generation product target (July 2025 material) was **256Gb (gigabits) per die, a 16-high stack of 512 GB, and 1.6 TB/s read bandwidth,** aiming at a footprint, power profile and stack height close to HBM4's. 256Gb is 32 GB, so 16-high gives 512 GB. The 1.6 TB/s is the target for one product, within the grade range of the standard above.
- Schedule: in August 2025, Sandisk set targets of first HBF samples in the second half of 2026 and samples of HBF-equipped AI inference devices in early 2027. Its investor day presentation of August 2026 (Sandisk Investor Day 2026, HBF presentation PDF) put the tape-out of the first HBF memory die and 2027 samples of the first HBF inference product on its roadmap, and did not repeat the target of memory samples in the second half of 2026.
- Technology base: Sandisk's BiCS NAND and CBA (CMOS directly Bonded to Array, a structure that bonds the CMOS circuit wafer directly to the cell-array wafer) + SK hynix's HBM packaging and 3D stacking capabilities.

The structure of the trade — what you give and what you get

ITEM	HBM4 (FOR REFERENCE)	HBF (SPECIFICATION RELEASED 2026-08)
Stack capacity	Tens of GB (volume 12-high 24–36 GB, standard maximum 64 GB)	**Up to 512 GB (8-high, 16-high)**
Read bandwidth	About 2 TB/s by the standard, up to 3.3 TB/s in volume products	Three grades, about 0.4–3.0 TB/s
Write behavior	Symmetric, unrestricted	**Asymmetric — slow writes, block erase required**
Endurance	Effectively unlimited	**Finite P/E cycles**
Latency	Low	High
Use	Training and inference in general	**Mainly keeping weights resident during inference**

The key point: HBF is not something that replaces HBM; it **adds one level to the memory hierarchy.** The intended arrangement keeps large data that rarely changes (model weights, inactive MoE experts, long-context caches) in HBF and keeps actively updated data in HBM.

Mixing the two levels — a simple model

Suppose that in one decoding step a fraction f of the bytes read comes from HBF and the remaining $1 - f$ from HBM. If the two are read one after the other, the time taken is the sum of the two shares, so the effective bandwidth is

$$B_{\text{eff}} = \frac{1}{(1-f) \,/\, B_{\text{HBM}} + f \,/\, B_{\text{HBF}}}$$

and the capacity available is the sum of the two levels, $C_{\text{HBM}} + C_{\text{HBF}}$.

Worked example. Pair one HBM4 stack (up to 3.3 TB/s as announced by Samsung, 12-high 36 GB) with one HBF stack (512 GB). For HBF bandwidth, use the bottom of the published specification range, 0.4 TB/s, as an assumed value. If only 10% of the bytes read come from HBF ($f = 0.1$),

$$B_{\text{eff}} = \frac{1}{0.9 \,/\, 3.3 + 0.1 \,/\, 0.4} \approx 1.91 \text{ TB/s}$$

which is 58% of reading from HBM alone. At the same f, assuming the top grade of 3.0 TB/s for HBF gives about 3.27 TB/s, almost no loss. Capacity is 36 + 512 = 548 GB either way. Let the share read from the slow level grow even a little and that level sets the overall speed. So HBF's place is data for which f can be kept small — for example, in MoE, the expert weights not used in the current step: mostly just held, read only now and then (Exercise 4).

This model assumes the two levels are read in sequence. It leaves out reading both at once so the reads overlap, and moving needed data into HBM ahead of time (prefetching), and it folds NAND's long latency into a single bandwidth number. All bandwidth values are announced or assumed values for a single stack.

The figure below shows the worked example above: $f = 0.1$, HBM4 at 3.3 TB/s and 36 GB, HBF at 0.4 TB/s (assumed) and 512 GB. In the screen edition you can change f, HBM bandwidth and capacity, and HBF bandwidth (within the published range of 0.4–3.0 TB/s).



Figure 3-1. If 10% of the bytes read come from HBF (0.4 TB/s, assumed), the effective bandwidth mixed with an HBM4 stack (3.3 TB/s) is about 1.91 TB/s, 58% of reading from HBM alone, and capacity is 548 GB. The band is the 0.4–3.0 TB/s range of published HBF specs. — In the HTML edition you can change the values here (HTML edition, Session 3, Section 6).

Open questions — what has not yet been verified

- NAND read latency is far longer than DRAM's. How many real workloads can offset it with bandwidth?
- How will write endurance (wear leveling) be managed at the accelerator level?
- Will the software stack (the policies that decide which data goes on which level) mature?
- How standardization proceeds and whether the technology is actually adopted — **as of October 2026 the first standard specification has been released but there are no volume products, and the performance figures are specification ranges and targets. Treat this as "an attempt in progress," not "an established fact."** Intel Optane (3D XPoint), an earlier attempt to fill the gap between DRAM and NAND, began winding down in the second quarter of 2022. A candidate for a new level does not necessarily take hold (appendix B).

> **ANOTHER ANSWER ON THE CAPACITY AXIS — CXL**
>
> If HBF is the path that attaches capacity close to the accelerator, CXL (Compute Express Link) is the path that adds memory on the server side. CXL is an open standard that borrows the physical and electrical interface of PCIe while keeping cache coherence between the CPU's memory space and the memory of attached devices. It is used to add memory to a single server (expansion) or to let several servers share a pool of memory (pooling). The CXL Consortium released CXL 4.0 on November 18, 2025, doubling the transfer rate from 64 GT/s to 128 GT/s. At FMS 2026, SK hynix demonstrated several servers moving and reusing KV cache through CXL pooled memory. In the hierarchy table of section 1, CXL memory belongs to the DRAM level. One x16 link works out to about 256 GB/s in one direction (128 GT/s × 16 ÷ 8, ignoring encoding overhead), around one-tenth of a single HBM4 stack. So it is a place not for data read on every token but for data fetched now and then, such as overflow KV cache.

---

7. PIM / PNM — taking computation to the memory

In the roofline formula $P = \min(P_{\text{peak}}, B \times I)$, the approaches so far have raised I (software) or raised B (HBM). A third approach is **not to move the data in the first place.**

- **PIM (processing-in-memory):** put simple compute units inside the memory die. Place a multiply-accumulate unit in each DRAM bank and the internal bandwidth (not limited by external pins) can be used directly. Samsung's HBM-PIM and SK hynix's AiM are among the attempts.
- **PNM (processing-near-memory):** put compute units near the memory (for example, on the base die or a buffer chip).

Why it is not yet mainstream

- A DRAM process is not a logic process. Putting complex logic on the same die worsens cost and yield.
- There is no programming model. The compilers and runtimes that decide which operations to send down to memory are immature.
- It is not general-purpose; the gain is large only for particular operations (for example, embedding lookup and GEMV).

Why it still deserves attention: the move to make the base die in a foundry logic process, as in Samsung's HBM4 (section 4), is a change that opens the door to PNM. **The boundary between memory and compute is physically blurring.**

---

8. Common misconceptions

Misconception 1: "HBM is just fast DRAM" → The cells themselves are not very different from ordinary DRAM. What differs is **the interface width and the packaging.** HBM is closer to a packaging technology than to a memory technology.

Misconception 2: "HBM is always better than GDDR" → It wins on absolute bandwidth and power efficiency, but it costs far more, its capacity is harder to scale, and it is in short supply. There is a reason gaming GPUs still use GDDR.

Misconception 3: "Increase HBM capacity and the memory problem is solved" → Capacity and bandwidth are different problems. And raising bandwidth does not settle things either. If P_{peak} stays the same and B grows, the ridge point $I^* = P_{\text{peak}} / B$ comes down (for example, the H100 from Session 2, 989 TFLOP/s ÷ 3.35 TB/s ≈ 295 FLOP/byte, becomes about 148 FLOP/byte if B doubles). When the ridge point comes down, workloads with lower arithmetic intensity also cross over to the compute-bound side, but a workload like decoding, whose arithmetic intensity is below even the new ridge point, stays bound by bandwidth.

Misconception 4: "DRAM keeps getting faster as it scales" → Bandwidth grows, but latency does not improve. If anything, there is pressure in the other direction: cell capacitors shrink and sensing becomes harder.

Misconception 5: "HBF will replace HBM" → Because of its write behavior and endurance, it adds a level rather than replacing one. And as of October 2026 only the first standard specification has been released; there are no volume products.

Misconception 6: "More layers increase capacity at linear cost" → The capacity of one good stack is proportional to its height (die capacity × number of dies). What is not proportional is the cost of getting that stack. Stack without screening and stack yield falls as y^n, so the dies needed for one good stack, n / y^n, grow faster than the height. Taller stacks also tighten the constraints of heat removal and stack height (package standards).

Misconception 7: "An HBM channel is like a mainframe I/O channel" → Only the name is shared. A mainframe I/O channel is a dedicated processor that handles input and output on the CPU's behalf. An HBM channel is a unit into which the stack's wide bus is divided, each part operating independently. Bus width = number of channels × channel width, so HBM3 is 16 channels × 64 bits = 1,024 bits and HBM4 is 32 channels × 64 bits = 2,048 bits (the channel width is the bus width the standard states divided by the channel count). GDDR has the same structure; in GDDR7, JEDEC increased the independent channels from 2 in GDDR6 to 4.

Misconception 8: "Refresh refills the charge that leaves during a read" → That is restore. Restore is caused by an access (the read); refresh is caused by time. Even cells that are not read must be refilled within a fixed period because of leakage. For DDR4, at or below 85°C, 8,192 REFRESH commands must be issued within 64ms (section 2).

Misconception 9: "Pin speed is stuck at a physical limit and cannot go higher" → It is not that signals cannot be sent faster; it is that sending them faster is expensive. The GDDR7 standard already provides 48 Gbps per pin (192 GB/s per device divided by 32 bits). What holds speed back is not an absolute limit but the channel loss that grows with speed, the pJ/bit driven up by the equalization circuits that compensate for it, and the shorter distance a signal can travel (section 3).

---

9. Quantitative exercises

Exercise 1 — Stack yield Vary die yield y over 0.95 / 0.98 / 0.995, calculate the stack yield for 8-, 12- and 16-high stacks, and find the number of good dies discarded without KGD screening. **The answer to "why do HBM makers invest so much in testing?" comes out in numbers.**

Exercise 2 — Bandwidth design To reach a target bandwidth of 4 TB/s with (a) a 1,024-bit bus and (b) a 2,048-bit bus, what must the per-pin transfer rate be? Discuss qualitatively how the difference in pJ/bit between the two affects system power.

Exercise 3 — Allocating the memory budget When serving a given model, how should the available memory be split between weights and KV cache to maximize throughput? A larger batch grows the KV cache; quantization shrinks the weights. Set it up as a problem of finding the optimum.

Exercise 4 — Judging fitness for HBF Set up criteria for judging which workloads would benefit from HBF. (Hint: read/write ratio, reuse distance, latency sensitivity.)

---

10. Discussion questions

1. Reconstruct the shape of HBM from its physical constraints. If there had been no signal-integrity problem, would HBM exist?
2. How might the move to make HBM4's base die in a logic foundry process (Samsung has done it with its own foundry's 4nm process) change the competitive structure of the memory industry?
3. If the memory hierarchy is subdivided further into SRAM–HBM–DRAM–HBF–SSD, what does software have to do? Can that decision be automated?
4. What must be solved first for PIM to become practical? Is it a hardware problem or a software problem?
5. If the bandwidth race keeps depending on pJ/bit improvements, where is the physical limit? Could optical interconnect be the answer? (Continued in Session 7.)

---

11. Further study

- **Memory systems textbook:** Jacob, Ng & Wang, *Memory Systems: Cache, DRAM, Disk* (Morgan Kaufmann, 2007) — the standard reference on DRAM timing and structure
- **DRAM research:** lecture materials and papers from the home page of Prof. Onur Mutlu (ETH Zürich) and the SAFARI Research Group — the most systematic body of work on DRAM latency, refresh and PIM, with lecture videos available online. **Recommended for readers who want to go deeper into DRAM**
- **HBM standards:** JEDEC JESD235 (HBM), JEDEC JESD270-4 (HBM4) — the standard documents themselves (as of October 2026 the HBM4 standard page carries the revision JESD270-4A). For summaries, the technical blogs of Rambus and Synopsys are accessible
- **The signal-speed standard:** JEDEC JESD239 (GDDR7) — the current form of path ①, raising pin speed with PAM3
- **HBF:** Sandisk press release on collaborating with SK hynix on standardization (2025-08-06), SK hynix, first HBF standard specifications (2026-08-04), IEEE Spectrum, "The Memory in Your Thumb Drive Could Fix AI's Big Problem" (2026-07-14), OCP workstream documents
- **PIM:** ISSCC and Hot Chips presentations on Samsung's HBM-PIM and SK hynix's AiM
- **Korean technology trends:** ETRI, *Electronics and Telecommunications Trends* vol. 41 no. 3, "Key Technologies and Market Trends in Artificial Intelligence (AI) Memory Semiconductors" (in Korean)

Self-study appendix A — solutions to the quantitative exercises

Solution to Exercise 1 — Stack yield

▾ Solution

Set bonding yield to 1 and look only at die yield: $Y_{\text{stack}} = y^n$.

DIE YIELD y	8-HIGH	12-HIGH	16-HIGH
0.95	66.3%	**54.0%**	44.0%
0.98	85.1%	78.5%	72.4%
0.995	96.1%	94.2%	92.3%

Stop here and you have seen only half. The real question is "if you stack without screening, how many perfectly good dies are thrown away with the bad ones?"

The expected number of good dies in one failed stack is

$$E[\text{good dies} \mid \text{stack fails}] = \frac{ny - ny^n}{1 - y^n}$$

Substituting $n = 12$, $y = 0.95$:

$$\frac{12(0.95) - 12(0.5404)}{1 - 0.5404} = \frac{11.40 - 6.485}{0.4596} = 10.7 \text{ dies}$$

Applying the same formula to all nine combinations gives the following. In each cell, the first value is the expected number of good dies in one failed stack, and the value in parentheses is the expected number of good dies discarded when making 1,000 stacks, 1,000 ($ny - ny^n$).

DIE YIELD y	8-HIGH	12-HIGH	16-HIGH
0.95	6.8 (2,293)	**10.7 (4,916)**	14.6 (8,158)
0.98	6.9 (1,034)	10.9 (2,343)	14.8 (4,099)
0.995	7.0 (274)	11.0 (641)	15.0 (1,153)

As die yield rises, the total number of good dies discarded (in parentheses) drops sharply, but the number of good dies in one failed stack actually approaches $n - 1$. The rarer defects become, the more a failed stack is one with "only one bad die."

Suppose you make 1,000 12-high stacks.

- 12,000 dies go in, of which 11,400 are good
- 540.4 stacks (about 540) come out whole → 6,484.3 good dies survive (about 6,484)
- **4,915.7 good dies (about 4,916) are discarded — 43% of the good dies are wasted**

(Fractions remain because these are expected values. They multiply the unrounded stack yield 0.54036.)

What these numbers explain

1. **Why so much is invested in KGD (known good die) screening.** Perfect screening brings that 43% to 0. It recovers about 4,916 good dies per 1,000 stacks, so if one die is worth c, screening wins when its extra test cost is less than about $4{,}916c$ (include the stacking and bonding costs spent on discarded stacks and the threshold rises further). Real test costs and die prices are not part of this exercise.
2. **Why it is so hard to raise the number of HBM layers.** Going from 8-high to 16-high drops yield from 66% → 44% at $y = 0.95$. To double the height while keeping the same stack yield, die yield has to be pushed from 0.95 to around 0.975.
3. **Why HBM supply does not grow quickly even when more money is offered.** Section 4 showed that packaging capacity sets the ceiling on shipments. This calculation adds a different constraint on top of that: the yield learning curve. With low yield, the same capacity turns out fewer good stacks, and yield is hard to pull

forward with money the way equipment can be bought. This calculation has no capacity or expansion period as an input, so it cannot tell which of the two binds harder right now. This is what the movement of the bottleneck seen in Session 1 actually looks like.

Add bonding yield and it gets worse. Even with $y_{\text{bond}} = 0.999$, a 12-high stack has 11 bonds, so $0.999^{11} = 0.989$ is multiplied in.

What this calculation leaves out

- Die defects are treated as independent. Real defects tend to cluster on the wafer.
- "Perfect screening" is the extreme in which the KGD test misses no defect at all. Test cost and the miss rate are not included.
- The effect of repairing defects with redundancy is taken to be already included in the die yield y.

The figure below shows a sample of 100 stacks made without screening at die yield $y = 0.95$, 12-high. In the screen edition you can change die yield, stack height, bonding yield and whether dies are screened before stacking.

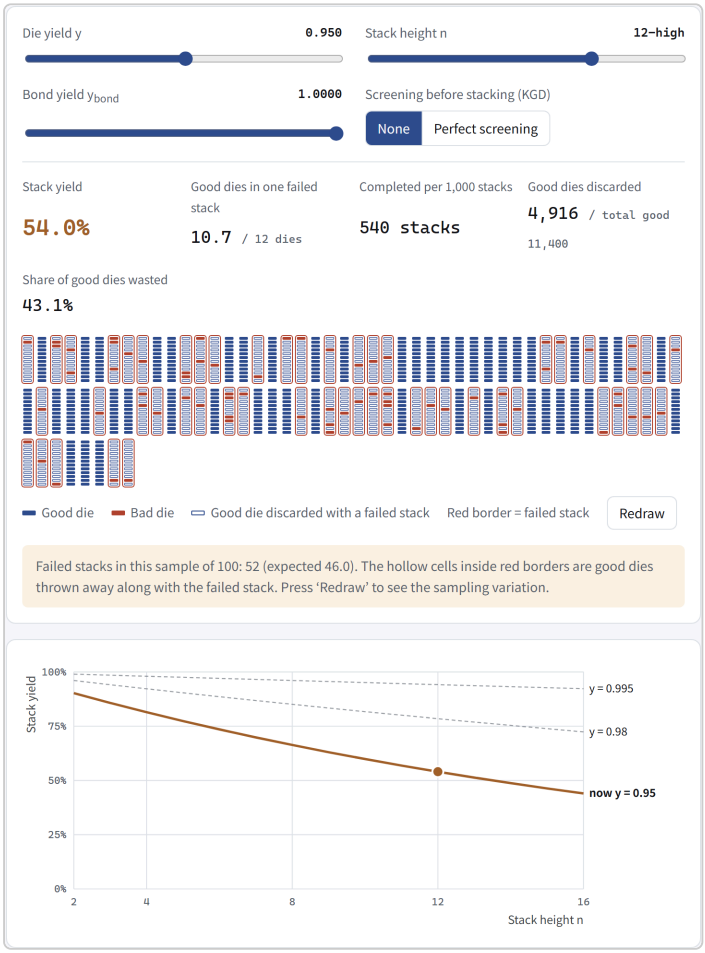


Figure 3-2. Stacking 12 dies with a die yield of 0.95 and no screening gives a stack yield of 54.0%, and each failed stack throws away 10.7 good dies on average. The bars are 100 randomly drawn stacks. — In the HTML edition you can change the values here (HTML edition, Session 3, Solution to Exercise 1).

Solution to Exercise 2 — Bandwidth design

▾ Solution

B = (bus width) × (transfer rate per pin). Target 4 TB/s $= 3.2 \times 10^{13}$ bit/s.

BUS WIDTH	REQUIRED PER-PIN TRANSFER RATE	HOW IT IS DONE
(a) 1,024 bit	$3.2 \times 10^{13} / 1024 =$ **31.25 Gbps**	SerDes territory. Equalization and CDR circuits required
(b) 2,048 bit	**15.6 Gbps**	Possible with a parallel interface

The pJ/bit difference and power

As the transfer rate rises, energy per bit grows not linearly but steeply, because equalization, clock and data recovery (CDR), driver strength and termination losses all pile on. As a rough sense:

APPROACH	ROUGH PJ/BIT	POWER AT 4 TB/S
High-speed SerDes (30 Gbps class)	3–5	**96–160 W**
Wide parallel (15 Gbps class)	0.5–1	**16–32 W**

3.2×10^{13} bit/s × 3 pJ/bit = 96 W. That is for **one** stack.

Conclusion: with 8 stacks on one accelerator, (a) at the low end of the range, 3 pJ/bit, takes about 770W (96W × 8) for memory alone, approaching 800W; at the high end, 5 pJ/bit, it is 1,280W. (b) at 0.5–1 pJ/bit takes about 130–260W. When the accelerator itself draws 700–1,000W, this difference changes the system design. **Power is why HBM uses "very many slow pins."**

The price: 2,048 pins cannot be planted along the edges of a die; they need an **area array** → microbumps → interposer → CoWoS. In Exercise 5 of Session 6 you count these wires yourself. This holds when all 2,048 pins are used, though. Like SPHBM4 at the end of section 3, cutting the signals to 512 and running each four times faster lets the stack sit on an organic substrate. Four times the HBM4 standard of 8 Gbps is 32 Gbps per signal, a speed similar to (a) in this exercise. It is a choice to avoid the interposer at the cost of carrying (a)'s pJ/bit burden (SPHBM4's actual power figures are not in the JEDEC press release).

What this calculation leaves out: the pJ/bit ranges are a rough sense, not vendor measurements. The relationship between pin speed and pJ/bit has not been turned into a formula either — real values differ by circuit and process.

The figure below assumes a target of 4 TB/s per stack, 8 stacks per accelerator, (a) 3 pJ/bit and (b) 0.75 pJ/bit. In the screen edition you can change the target bandwidth, the number of stacks and the pJ/bit of the two approaches.

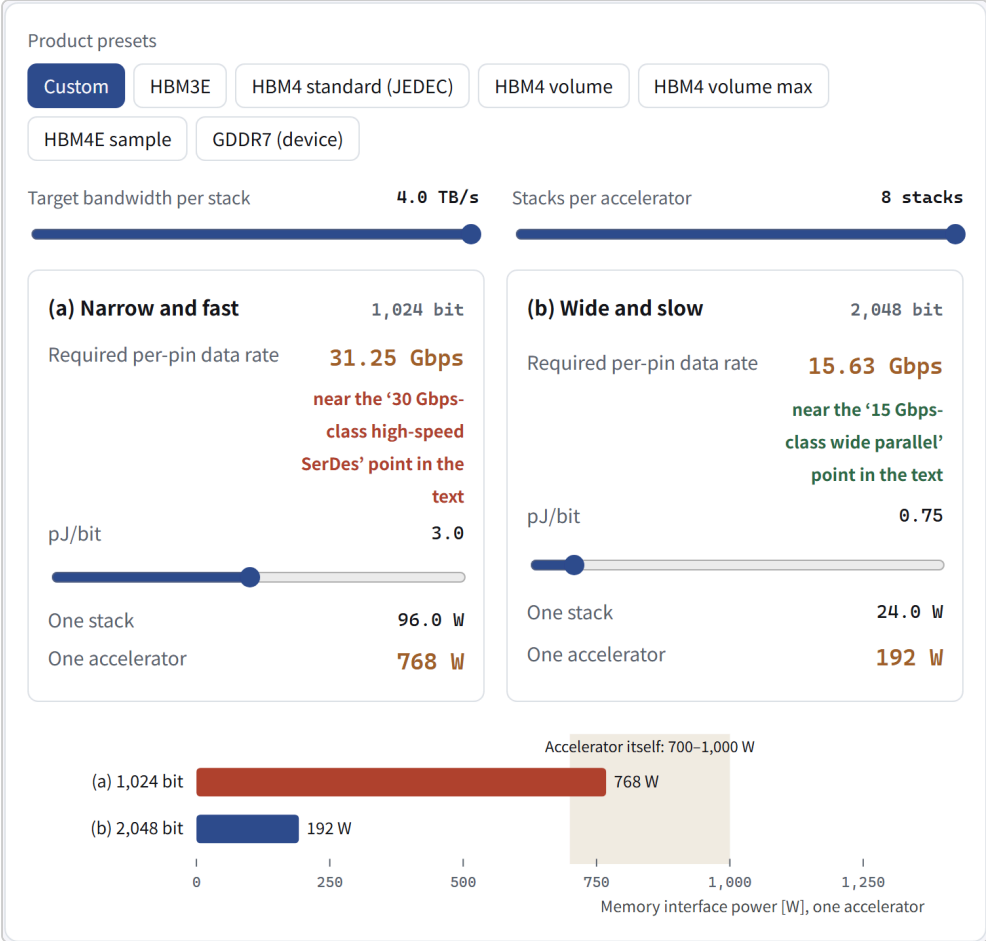


Figure 3-3. Delivering 4 TB/s per stack over a 1,024-bit bus takes 31.25 Gbps per pin; over a 2,048-bit bus, 15.6 Gbps. With energy per bit set to 3 and 0.75 pJ/bit, memory-interface power for 8 stacks approaches 800 W for (a) and is about 190 W for (b) (about 130–260 W over the 0.5–1 pJ/bit range). — In the HTML edition you can change the values here (HTML edition, Session 3, Solution to Exercise 2).

Solution to Exercise 3 — Allocating the memory budget

▾ Solution

Set the problem up as a formula. With total memory M, weights $P \cdot b$ and KV cache per sequence k: the weights must fit first ($M \geq P \cdot b$, or the model cannot be served), so the batch is the number of sequences that fit in the remaining capacity, rounded down to an integer.

$$B_{sz} = \left\lfloor \frac{M - P \cdot b}{k} \right\rfloor, \qquad \text{throughput} \lesssim \frac{B}{P \cdot b} \times B_{sz} \leq \frac{B\,(M - Pb)}{P\,b\,k}$$

The rightmost expression is the continuous ceiling without the rounding. The properties below are read from that continuous ceiling; the batches in the table are rounded down.

Look at it in terms of b: the numerator grows and the denominator shrinks — a double gain. That is why quantization has a far larger effect than intuition suggests.

Numerical check (70B, $M = 80$ GB, $k = 1.34$ GB, $B = 3.35$ TB/s)

PRECISION	WEIGHTS	BATCH	THROUGHPUT CEILING	MULTIPLE
fp16	140 GB	—	Does not fit	—
INT8	70 GB	7	335 tok/s	×1
INT4	35 GB	33	**3,159 tok/s**	**×9.4**

Halving the precision raised throughput not 2 times but 9.4 times. The doubling of the ceiling is multiplied by a 4.7-times larger batch, because the spare capacity grows from 10 GB → 45 GB.

Properties as an optimization problem

- b is not a free variable. It is the smallest value you can choose **under the constraint of accuracy loss.**
- Quantizing the KV cache separately shrinks k for a further gain. KV quantization is often more forgiving of accuracy than weight quantization, so in practice it is applied first.
- With longer sequences, $k \propto S$, so the batch collapses. This term is **why long-context services are expensive.**

What to take away: whenever someone says "add memory," always ask whether they mean bandwidth or capacity. Here it was **capacity** that governed throughput.

The same calculation can be tried in the figure at the end of the solution to Exercise 3 in Session 2. In the screen edition, change the sequence length and precision and watch how the maximum batch changes.

Solution to Exercise 4 — Judging fitness for HBF

▾ Solution

For NAND-based high-capacity memory to pay off, the workload must not touch NAND's weak points. Set up the criteria on four axes.

AXIS	FAVORS HBF	WORKS AGAINST HBF
Read/write ratio	Read-dominated (almost no writes)	Frequent writes — NAND is weak on write endurance and latency
Reuse distance	Long — not covered by an HBM cache	Short — it stays in HBM anyway
Access granularity	Large and sequential	Small and random
Latency sensitivity	Low, or **known in advance so it can be prefetched**	High and unpredictable

Good fits

- **MoE expert weights** — most are inactive and read-only. But the router chooses experts only after seeing the hidden state that enters that layer, so the decision for the same layer cannot be known in advance. For prefetching to work you need either to predict the next layer's experts from the previous layer's hidden state and read them early (reading that much later when the prediction misses) or a schedule that overlaps the computation of several layers and tokens to hide the read time. Given such mechanisms, this is counted as HBF's most promising use.
- **The less-read parts of a very long KV cache** — with dense attention, which looks at every earlier position, each new token reads the K and V of all earlier positions, so older tokens are not read less just because they are old. Access frequency actually falls when sliding-window or sparse attention is used, or a policy that reads only selected important positions, or when the KV cache belongs to a conversation that is waiting rather than being decoded now. Under those conditions it becomes a candidate for tiering.
- **RAG vector indexes and embedding tables** — large, read-only, batched access.
- **Offline batch inference** — there is no latency requirement at all.

Poor fits

- **Optimizer state during training** — writes happen at every step. NAND endurance becomes a problem immediately.
- **The hot path of interactive decoding** — the active weights needed for every token. Latency shows directly to the user.
- **Workloads dominated by small, random accesses** — amplification is severe on NAND, which is accessed page by page.

The one-line rule: "Is it read-only, is it large, and can you know in advance when it will be needed?" If all three hold, HBF; if any one fails, keep it in HBM.

Self-study appendix B — leads for the discussion questions

TYPE OF QUESTION	WHERE IT DIVIDES	LEADS
Will the HBM oligopoly last?	**The nature of the entry barrier**	If the barrier is patents, they expire; if it is capacity, money loosens it; but if it is the **yield learning curve,** only time loosens it. What Exercise 1 shows is the weight of the third type — where yields multiply, a small difference in die yield opens into a large difference in stack yield, and a latecomer can hardly skip the leader's accumulated learning with money. This calculation cannot tell which of the three barriers is largest right now
Will HBF replace HBM?	**Replacement vs. an added level**	It is more accurate to see it not as replacement but as **one more level.** Historically, memory hierarchy levels have never disappeared; they have only been added (registers → cache → DRAM → SSD). HBF is a candidate to fill the gap between DRAM and SSD in the order HBM–DRAM–HBF–SSD. But new levels have not always taken hold. Intel Optane (3D XPoint), which aimed at the same gap, began winding down in the second quarter of 2022. HBF faces the same test
Why doesn't PIM work yet?	**Ecosystem vs. device**	The barrier has two layers. One is hardware: putting logic into a DRAM process worsens cost and yield (section 7). The other, emphasized here, is the **programming model.** Sending computation to memory requires compilers, frameworks and kernels all to change, and the lock-in in section 7 of Session 2 operates exactly here. It is also why hyperscalers can try PIM in their in-house chips
Are memory prices normal?	**Cycle vs. structural change**	The same question as discussion question 5 in Session 1. Here, though, there is one more piece of evidence — HBM eats into ordinary DRAM capacity, so **a path exists by which AI demand reduces the supply of ordinary memory.** This is a structure that past cycles did not have

Self-study appendix C — key terms

TERM	ONE-LINE DEFINITION
Bandwidth / capacity	The amount moved per second / the amount held. **The two axes to keep apart throughout this session**
HBM (high bandwidth memory)	Memory that stacks DRAM dies vertically and connects them through a wide bus
HBF (high bandwidth flash)	A proposal to stack NAND the same way to reach high capacity. First standard specification released in August 2026; no volume products yet
CXL (Compute Express Link)	An interconnect standard that keeps cache coherence between the CPU and devices on top of the PCIe physical layer. Used for server memory expansion and pooling
TSV (through-silicon via)	Vertical wiring that runs through silicon
KGD (known good die)	A good die verified before stacking
Microbump	A tiny solder bump that joins one die to another

TERM	ONE-LINE DEFINITION
Hybrid bonding	Joining copper to copper directly without bumps. Better in both pitch and thermal resistance. For HBM, as of October 2026, a candidate for the next generation
MR-MUF / TC-NCF (mass reflow molded underfill / thermal compression non-conductive film)	The stacking methods of volume HBM. Fill and cure a liquid protective material all at once after stacking (SK hynix) vs. lay a film on each layer and thermocompression bond it (Samsung)
Interposer	An intermediate substrate on which several dies sit and through which they are connected
Refresh	Rewriting DRAM capacitors at a set interval before their charge leaks away. Caused by time, and different from the restore right after a read
Bank	A group of cell arrays inside a DRAM die that can be opened and closed separately. The unit of internal parallel access
Channel	A unit into which a memory interface is divided, each part operating independently. Bus width = number of channels × channel width (HBM4: 32 × 64 bits). Not the same thing as a mainframe I/O channel
pJ/bit (picojoule per bit)	Energy to transfer 1 bit. The practical constraint on bandwidth design
PIM / PNM (processing in / near memory)	Placing compute units inside / beside the memory
Stack yield	The probability that the whole stack works. Falls steeply as y^n
Memory tiering	Placing data on different media according to access frequency

---

Self-study appendix D — self-assessment

☐ I can explain with an example that bandwidth and capacity are **different constraints** → the whole session ★

☐ In $B =$ bus width × transfer rate, I can explain **in terms of power** why the industry chose "wide and slow" → Exercise 2 ★

☐ I can estimate the stack yields of 8-, 12- and 16-high stacks almost in my head → Exercise 1

☐ I can say **in numbers why KGD screening is essential** → Exercise 1 ★

☐ I can explain why HBM supply does not grow quickly with money → Exercise 1

☐ I can name three criteria for judging which workloads suit HBF → Exercise 4

☐ I can explain why MoE is a promising use of HBF → Exercise 4

☐ I can describe the barriers holding PIM back as process (hardware) and programming model (ecosystem) → section 7

Before the next session

With these three sessions we have seen **what the problem is.** From Session 4 on, we turn to **who makes it and how.**

QUESTION THIS SESSION LEAVES	ANSWER IN A LATER SESSION
If stack yield matters that much, what sets yield in the first place?	**Session 4** — the Poisson model and defect density
What exactly are the interposer, TSV and bonding processes?	**Session 6** — advanced packaging
How is room made for 8 stacks?	**Session 6** Exercises 3 and 5
If HBM runs that hot, how is it cooled?	**Session 7** — the thermal resistance chain

---

Sources

Introductory material chosen by the study curriculum (the "AI memory semiconductors" item)

CURRICULUM SUBTOPIC	LINK	HOW THIS BOOK USES IT
Types of memory	Semiconductor Network, "The Evolution of Memory Semiconductors in the AI Era… from Flash, SRAM and DRAM to HBM" (in Korean, 2026-06-29)	Overview of memory types. Basic material for the SRAM/DRAM/NAND distinction in section 1. The same article ran in *Semiconductor Network*, July 2026 issue, p. 44
HBM	Rambus blog, "High Bandwidth Memory (HBM): Everything You Need to Know"	The standard English explainer of HBM structure and specifications by generation. **Corresponds directly to section 4; its figures are good, so it is recommended for presentations**
HBF	HyperAccel tech blog, "Memory, the Essential Commodity of the AI Era — Understanding Memory, Part 1: Understanding HBF" (in Korean)	Explainer of the HBF concept. Basis for section 6
HBF	Daum News, "[Analysis] HBM? DRAM? The real demand is 'NAND'… the 'warehouse war' of the AI era" (in Korean)	Korean press coverage of HBF. Industry context
HBF	Growth Research, "[HBF Industry Report] The AI bottleneck was memory, not the GPU: the AI semiconductor revolution HBF will bring" (2025-11-17, Han Yonghui, Koo Minjun, Choi Hyein; in Korean) PDF	Industry outlook report on HBF. The market side of section 6. **It is written from an investment perspective, so read its forecasts critically.** No public posting location was found, so no link is given
AI memory technology trends	ETRI, *Electronics and Telecommunications Trends* vol. 41 no. 3 (2026), Hong Areum, Lee Jaeseong, Yeon Juwon, "Key Technologies and Market Trends in Artificial Intelligence (AI) Memory Semiconductors", pp. 15–27 (in Korean)	A Korean academic review. **A systematic Korean-language survey of AI memory as a whole, suitable as a companion text for this entire chapter**

Verified (2026-08-02, checked against several sources)

- HBM4 = JEDEC JESD270-4, **published April 2025.** 2,048-bit interface, 32 independent channels (2 pseudo-channels each), up to about 2 TB/s per stack, 4- to 16-high, 24Gb/32Gb dies, 64 GB per cube at 32Gb 16-high, backward compatibility with HBM3 controllers, DRFM, core voltage 1.0/1.05V and VDDQ 0.7–0.9V — checked against the JEDEC press release, TrendForce (2025-04-21) and Tom's Hardware
- HBF standardization: Sandisk–SK hynix collaboration announced **2025-08-06**; OCP workstream begun **2026-02-25.** Target specification 256Gb per die / 512 GB at 16-high / read about 1.6 TB/s, aiming at a footprint similar to HBM4's. First samples targeted for the second half of 2026 — checked against Sandisk and SK hynix press releases, IEEE Spectrum and Tom's Hardware. The per-die capacity and footprint wording confirmed against the original Sandisk HBF fact sheet ("256Gb per die," "closely matching")
- The three-company HBM oligopoly and SK hynix's lead by revenue (IDC figures for Q1 2026, Session 1 section 4) — verified. The order of second and third changes by generation and is not given

Verified (2026-10-08, checked against primary sources)

- HBM4 volume production: SK hynix, "SK hynix Completes World's First HBM4 Development and Readies Mass Production" (2025-09-12) — 2,048 I/O, over 10 Gbps, 1b DRAM, Advanced MR-MUF. Samsung Electronics, "Samsung Ships Industry-First Commercial HBM4 With Ultimate Performance for AI Computing" (2026-02-12) — 11.7 Gbps (up to 13 Gbps), up to 3.3 TB/s per stack, 12-high 24–36 GB, 1c DRAM and 4nm logic. Samsung Electronics, "Samsung Electronics Begins Shipment of Industry-First HBM4E Samples" (2026-05-29) — 14 Gbps (up to 16 Gbps), up to 3.6 TB/s per stack, 12-high 48 GB, "Samsung Foundry's 4nm logic base die"
- HBM4 standard: JEDEC JESD270-4 press release (2025-04-16) — 8 Gbps, 2,048 bits, up to 2 TB/s, 32 channels with 2 pseudo-channels each, VDDQ 0.7/0.75/0.8/0.9V and VDDC 1.0/1.05V, backward compatibility with HBM3 controllers, DRFM, 4-, 8-, 12- and 16-high with 24Gb and 32Gb dies (64 GB at 32Gb 16-high)
- NVIDIA Rubin: NVIDIA technical blog, "Inside NVIDIA Rubin GPU Architecture: Powering the Era of Agentic AI" (2026-07-21) — up to 288 GB of HBM4, 12-high stacks, up to 22 TB/s (2.8 times Blackwell). The post does not give the stack count; the 8 in the text is a back-calculated estimate
- HBF: SK hynix, first HBF standard specifications (2026-08-04) — released through the OCP, 8-high and 16-high, up to 512 GB, three grades of about 0.4–3.0 TB/s, UCIe, Google and Tenstorrent participating. SK hynix, FMS 2026 roundup (2026-08-07) — HBM, DRAM, NAND (HBF), SSD hierarchy; CXL pooled-memory demonstration. Sandisk standardization collaboration press release (2025-08-06) — targets of first samples in the second half of 2026 and inference-device samples in early 2027. Sandisk HBF fact sheet (2025-07) — first generation read 1.6 TB/s, 256Gb per die, 512 GB at 16-high, CBA = CMOS directly Bonded to Array
- Stacking processes: SK hynix HBM4 press release (above) — Advanced MR-MUF. Samsung

Electronics, "Samsung Develops Industry-First 36GB HBM3E 12H DRAM" (2024-02-27) — Advanced TC NCF

- SPHBM4: JEDEC JESD330-4 press release (2026-07-13) — the same DRAM dies as HBM4, 512 data signals with 4:1 serialization, mounting on organic substrates
- GDDR7: JEDEC JESD239 press release (2024-03-05) — up to 192 GB/s per device, PAM3 (3 bits in 2 cycles), independent channels 2 in GDDR6 → 4 in GDDR7. The 48 Gbps per pin is a calculated value, dividing by 32 bits
- DRAM refresh: Micron 16Gb DDR4 data sheet Rev. H (2021-08), description of the refresh function and Table 52 — 8,192 times within 64ms at 0–85°C (tREFI 7.8125µs), 32ms at 85–95°C (3.90625µs), tRFC1 160ns at 2Gb, 260ns at 4Gb, 350ns at 8Gb. The link is a copy hosted by a distributor (Digi-Key)
- CXL: CXL Consortium press release on releasing CXL 4.0 (2025-11-18) — 64 GT/s → 128 GT/s. CXL Consortium overview presentation — use of the PCIe physical and electrical interface
- Optane: Intel Form 10-Q (quarter ended 2022-07-02) — original on SEC EDGAR. It states that in Q2 2022 Intel began the wind-down of its Optane memory business and recorded an inventory impairment of USD 559 million in cost of sales in the same quarter. The next quarter's 10-Q (quarter ended 2022-10-01, original) gives the same amount as the cumulative figure for Q1–Q3 (checked 2026-10-08)

Unconfirmed / reserved (not asserted in the text)

- **Measured performance of HBF** — as of October 2026 there are no volume products. The 512 GB, 0.4–3.0 TB/s and 1.6 TB/s in the text are all **specification ranges and targets,** not verified measurements. The bandwidth of the middle grade (Grade 2) could not be confirmed from public material
- **HBF sample schedule** — the August 2025 target was first samples in the second half of 2026. Sandisk's August 2026 investor day material gave only the tape-out of the first HBF memory die and 2027 samples of the first inference product (checked 2026-10-08). When memory samples will ship has not been confirmed in an official announcement
- **Exact stack bandwidth of HBM2E/HBM3/HBM3E** — pin speeds differ by vendor and product, so the figures in the table are rough ranges. Exact values need checking against each vendor's data sheet
- **When hybrid bonding will be used in volume HBM** — as of October 2026 we found no manufacturer announcement that it has been used in volume HBM. Micron's stacking method was not confirmed from primary material
- **Change in HBM4's command bus structure** — statements listing the separation of command and data buses as an HBM4 change could not be confirmed in the JEDEC press releases (the 2024-07 preview and the 2025-04 publication), so it was left out of the list of changes in section 4. The body of the standard was not consulted
- **Percentage shares by HBM maker** — sources vary widely; not asserted
- **The y values in the section 5 yield calculation** — actual makers' yields are trade secrets. The figures in the text are **assumed values to show how the calculation works,** not real

yields

- **Commercial maturity of PIM products** — at the prototype and limited-deployment stage; whether they will be adopted broadly is undecided

SESSION 4

Fabless, Foundry and the Value Chain

KEY TAKEAWAY

The division of labor in the semiconductor industry (fabless, foundry, IDM, OSAT) is not a matter of taste but a consequence of its cost structure. As the cost of building a fab grew exponentially, manufacturing became an oligopoly of a few firms governed by economies of scale, while design became an asset-light field with many competitors. And nearly every economic judgment in this industry comes down to one formula: **yield $Y = e^{-AD_0}$ — as die area grows, yield falls exponentially.** This formula is the case for splitting chips into chiplets, the basis of how foundries price their work, and the reason advanced packaging exists (Session 6). "3nm process" is a marketing name, not a physical dimension. What you should compare is transistor density and cell structure.

PRIMARY SOURCES

Samsung Semiconductor Newsroom, "Foundry? Fabless? The Semiconductor Ecosystem at a Glance!" (in Korean) · ScienceON JAKO202470953293995, "Trends in Supporting Technologies for Advanced AI Semiconductors and Compilers" (in Korean) · ScienceON JAKO202369265025811, "Technical Trends in Hyperscale Artificial Intelligence Processors" (in Korean) · ScienceON ASTI00673636, "Market Insight 336: AI Application-Specific Semiconductors (AI ASIC)" (in Korean) · Winspec, "What Kinds of Semiconductor Processes Are There? Front End and Back End Compared" (in Korean) · TSMC Open Innovation Platform® · Samsung Foundry SAFE™ program · Stapper et al. (1980), doi:10.1147/rd.243.0398 · TSMC 2nm (N2) technology page

Learning objectives

1. Distinguish the roles of fabless companies, foundries, IDMs, OSATs and IP vendors, and how each makes money.
2. Explain the whole chip-design pipeline by mapping it onto the software development process.
3. Set up the Poisson yield model and calculate the number of good dies per wafer for different die sizes.
4. Know which metrics you actually need to check behind the label "3nm."
5. Decide quantitatively whether to develop an ASIC.

0. Self-study guide — for readers working alone

> **THIS SESSION BUILDS THE SECOND SHARED TOOL**
>
> If Session 2 built the roofline, this session builds the **yield formula**. The single line $Y = e^{-AD_0}$ governs all of Session 5 (why every process step has to be six sigma), Session 6 (why chiplets) and Session 7 (why supply does not grow). **Do Exercises 1 and 2 by hand.** Reading a table and putting numbers into the exponential yourself leave different things behind.

What carries over from earlier sessions

FROM EARLIER SESSIONS	IN THIS SESSION
The three layers of the value chain in Session 1	Sharpens that division in terms of **process and cost**
The ASIC break-even N^* in Session 2	Actually calculates c_A (ASIC unit cost) with the yield formula
Stack yield y^n in Session 3	Where that y comes from — defect density and die area

Prerequisite check

CONCEPT	ONE-LINE ANSWER	NEEDED FOR
Poisson distribution	The number of rare events in a fixed interval. $P(X = k) = e^{-\lambda}\lambda^k / k!$	★ Required (you only use $k = 0$; where that formula comes from is in section 5)
Gamma distribution	A continuous distribution over positive reals. Here it describes the variation in defect density	Section 5, advanced part only
Negative binomial distribution	The distribution you get when the Poisson mean is itself shaken by a gamma	Section 5, advanced part only
Die	One chip cut from a wafer	★ Required
Reticle limit	The largest area a lithography tool can expose in one shot (about 858 mm²)	Solution to Exercise 1 and Exercise 5 of Session 5

> **If you have a statistics background:** section 5 of this session is a rare case where the first chapter of an undergraduate probability course becomes an industry's cost structure as it stands. Exercise 3 (Poisson vs. negative binomial) reduces to one line of Jensen's inequality — see the solution in the appendix.

Reading order (110 minutes recommended)

STEP	SECTION	TIME	NOTE
1	Sections 1–4 (division of labor)	20 min	A refinement of Session 1. Read quickly
2	**Section 5, the economics of yield**	**35 min**	**The heart of this session.** Do Exercises 1 and 2 right here
3	Exercises 1, 2 and 3	25 min	Calculator required
4	Section 6, choosing among ASIC, FPGA and the rest	15 min	
5	Exercises 4 and 5 + self-assessment	15 min	Exercise 4 is a preview of Session 6

Never skip: the table "Getting a feel for the numbers" in section 5. Looking at that table, you should be able to explain from the properties of the exponential function **why yield halves when area grows 16-fold.**

A common sticking point — why the GDPW formula looks the way it does

$$\text{GDPW} \approx \frac{\pi d^2 \,/\, 4}{A} - \frac{\pi d}{\sqrt{2A}}$$

There is nothing to fear here. Read the two terms separately.

- **First term** = wafer area ÷ die area. Simply "how many fit."
- **Second term** = a correction for edge loss. When you fill a disc with squares, some dies straddle the rim and get cut off. The second term estimates their number as **wafer circumference ÷ die diagonal.** If a square die has side $\sqrt{A}$, its diagonal is $\sqrt{2A}$, which gives $\pi d \,/\, \sqrt{2A}$. Take the side $\sqrt{A}$ as the denominator instead and the loss grows by a factor of $\sqrt{2}$; that one constant changes the result visibly (table below).

In short, "divide by area, then subtract what is thrown away at the rim" is all there is to it. The larger the die, the greater the weight of the second term — **a big chip loses twice, once on yield and once at the rim.**

This formula is an approximation too. Count directly and the difference shows. The conditions are a 300 mm wafer, square dies, zero edge exclusion and zero scribe-line width. Lay a grid of dies over the wafer and count only the dies whose four corners all fall inside the circle of radius 150 mm. We shifted the origin of the grid horizontally and vertically in steps of 1/20 of a side, counted at 400 positions, and took the mean and range. The result:

DIE AREA	COUNTED DIRECTLY: MEAN (MIN–MAX)	$\sqrt{2A}$ FORMULA	$\sqrt{A}$ FORMULA
100 mm²	647.8 (640–657)	640.2 (−1.2%)	612.6 (−5.4%)
400 mm²	147.8 (144–151)	143.4 (−3.0%)	129.6 (−12.3%)
800 mm²	68.2 (66–71)	64.8 (−5.0%)	55.0 (−19.3%)

The percentages in parentheses are differences from the directly counted mean. The $\sqrt{2A}$

formula comes out 1–5% below the mean; the $\sqrt{A}$ formula, 5–19% below. The error grows with die size, and the count also depends on where the grid is placed (66 versus 71 at 800 mm^2). Align one die corner with the wafer center and you get 648, 148 and 68 dies at 100, 400 and 800 mm^2, almost the same as the mean. A real wafer has edge exclusion and scribe lines, so even the directly counted values shrink below these. For a comparison of several approximation formulas, see de Vries (2005), doi:10.1109/TSM.2004.836656.

---

1. Getting the intuition — why design and manufacturing split apart

The explosion of fixed costs

The cost of making a chip comes in two lumps.

- **Fixed costs (NRE, non-recurring engineering):** design labor, EDA tool licenses, IP licenses, **the cost of making the mask set**, verification and prototypes. Paid once, regardless of volume.
- **Variable costs:** wafers, packaging, test. Paid per unit.

On top of these sits a far larger fixed cost: **building the fab.** One state-of-the-art fab costs tens of billions of dollars. To recover a fixed cost of that size, the fab has to run 24 hours a day with enormous volumes flowing through it.

Conclusion: no company can fill a fab with its own chips alone. So two paths emerged.

1. **IDM (integrated device manufacturer):** when a company's own product volume is large enough (as with memory, standard products made in bulk). Samsung Electronics, SK hynix, Micron, Intel.
2. **The foundry–fabless division of labor:** gather the designs of many companies and fill one fab with them. TSMC established this model.

> **Economics frame:** an industry with large fixed costs and low marginal costs tends toward natural monopoly. Semiconductor manufacturing adds **a learning curve (yield rises as cumulative output grows)** on top, so the leader keeps gaining ground. What keeps latecomers from catching up is capital, and beyond capital, **accumulated experience.**

---

2. The players and their roles

TYPE	WHAT IT DOES	REPRESENTATIVE FIRMS	CHARACTERISTICS
Fabless	Design only; manufacturing is outsourced	NVIDIA, AMD, Broadcom, Qualcomm, Apple, MediaTek	Asset-light, high margins, fierce competition
Foundry	Manufacturing only; no design	TSMC, Samsung Foundry, Intel Foundry, GlobalFoundries, UMC, SMIC	Extremely high fixed costs; keeping customer secrets is everything
Design house (design services)	Implements and verifies a fabless company's logical design physically for a particular foundry process; sometimes also handles manufacturing, assembly and test on the client's behalf	Each foundry has an accredited partner network. TSMC has the DCA (Design Center Alliance, chip implementation services) and the VCA (Value Chain Alliance) inside OIP; Samsung has the Design Service Alliance (DSP, VDP) inside SAFE™	Works to the process and IP of the foundry it partners with. The route by which system companies and start-ups reach volume production
IDM	Design + manufacturing	Samsung, SK hynix, Micron, Intel, TI	Common in memory and analog
OSAT	Outsourced packaging and test	ASE, Amkor, JCET, PTI	Traditionally low value-added, but rising in standing with advanced packaging
IP vendor	Licenses design blocks	Arm, Synopsys, Cadence, Imagination	Royalty model
EDA	Design automation tools	Synopsys, Cadence, Siemens EDA	Three-firm oligopoly, extremely high barriers to entry
Equipment	Manufacturing tools	ASML, Applied Materials, Lam Research, TEL, KLA	Effectively an oligopoly or monopoly in each process
Materials	Wafers, gases, photoresist and more	Shin-Etsu, SUMCO, JSR, TOK, Linde	Large share held by Japanese firms

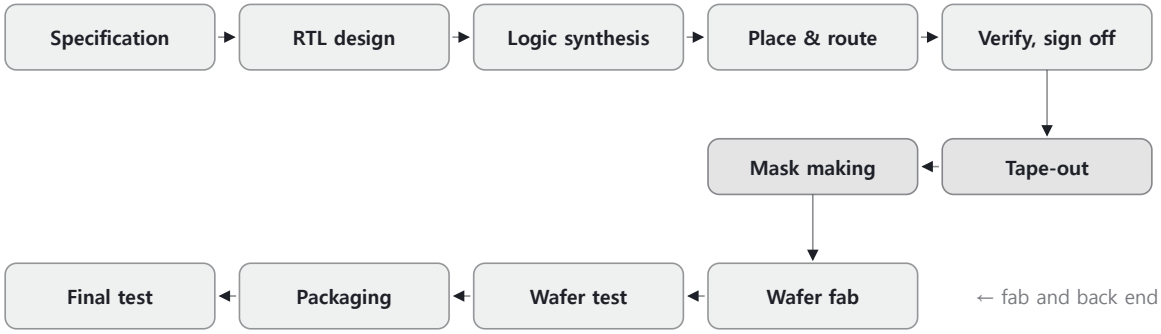
Where the boundaries blur (2026)

- **Memory makers need logic processes** — the HBM4 base die is made on a logic foundry process. In April 2024, when SK hynix signed a memorandum of understanding with TSMC on HBM4, it announced that it "plans to adopt TSMC's advanced logic process for HBM4's base die" and that the two companies "will first focus on improving the performance of the base die." Collaboration between memory makers and foundries becomes a necessity.
- **Foundries expand into packaging** — TSMC's CoWoS and SoIC. Foundries are absorbing territory that traditionally belonged to OSATs. The reason is plain: **advanced packaging decides performance, so it has to be optimized together with the front end.**
- **Hyperscalers become fabless companies** — Google, Amazon, Meta and Microsoft design their own chips. Most of them, though, work with design partners such as Broadcom and Marvell.
- **IDMs open foundries** — Intel Foundry, Samsung Foundry. The question of trust (can you

hand your design to a competitor?) is a structural problem.

---

3. The chip-design pipeline — mapped onto software development



Mapping table

CHIP-DESIGN STAGE	SOFTWARE COUNTERPART	WHAT IT DOES	WHAT PHYSICALLY EXISTS AT THAT STAGE
Specification (architecture)	System design	Sets performance, power and area (PPA) targets	Documents
RTL design (Verilog/VHDL)	**Writing source code**	Describes behavior at the register-transfer level	Code files
Functional verification (simulation, formal)	Testing, CI	Checks that the logic is correct	Simulation result files
Logic synthesis	**Compiling**	RTL → gate-level netlist of standard cells	Netlist file
Place and route (P&R)	Linking + layout optimization	Puts cells in real positions and wires them	Layout data
Timing/power/signal verification (STA etc.)	Profiling, optimization	Does each signal arrive within the clock period?	Verification reports
DRC / LVS	Lint + static analysis	Checks for manufacturing-rule violations and layout-versus-schematic agreement	Layout data that has passed verification
Tape-out	**Release (irreversible)**	Hands the data over for mask making	One set of layout files. Still nothing but data
Mask making	Producing distribution media	A mask set for a leading-edge node costs on the order of tens of millions of dollars and takes weeks	**The first physical object** — a bundle of photomasks for the process layers (the mask set)
Wafer fabrication	Running the build	Takes months	Wafers with the circuits patterned on them
Wafer test (EDS/probe)	Integration testing	Pass/fail judgment per die, repair	A wafer with pass/fail recorded for each die (wafer map)

CHIP-DESIGN STAGE	SOFTWARE COUNTERPART	WHAT IT DOES	WHAT PHYSICALLY EXISTS AT THAT STAGE
Packaging and final test	Packaging, deployment	Session 6	Cut dies → chips in packages

Read the fourth column from top to bottom: until tape-out, the chip **exists only as files.** What goes to the foundry at tape-out is a layout file, and the long-standing format is the GDSII stream. SEMI has standardized OASIS (SEMI P39) as its successor format. This column is why the software analogy ends at tape-out. Below that line, fixing anything means making the object again.

Two ways it differs decisively from software

1. **There are no hotfixes.** If a bug turns up after tape-out, you remake the masks (a respin) and spend months and tens of millions of dollars again. That is why verification takes more than half of the total development effort. **"The cost of one bug" differs from software by orders of magnitude.**
2. **Physics intervenes.** The same RTL can perform very differently depending on the result of place and route. Timing closure (making every path meet the clock period) is the main pain of late development.

There is one more place where the "RTL = source code" analogy breaks. RTL statements are not instructions executed from top to bottom; they describe **circuits that exist at the same time.** Three `assign` statements do not add three times; they place three adders side by side, and an `if` is not a branch but becomes a multiplexer that picks one of two inputs. What people coming from software most often get wrong is the nonblocking assignment (`<=`).

```
always @(posedge clk) begin
  q1 <= d;
  q2 <= q1;
end
```

Suppose that just before the clock edge `d = 1` , `q1 = 0` and `q2 = 0` . Read as top-to-bottom execution, `q1` and `q2` should both be 1 after the clock. In fact `q1 = 1` and `q2 = 0` . A nonblocking assignment reads all the right-hand values first at the clock instant and then updates the left-hand sides together (the nonblocking procedural assignment provisions of IEEE Std 1364-2005; that standard has since been superseded by IEEE Std 1800, SystemVerilog). Swapping the two lines gives the same result. This code is not two instructions but a picture of a two-stage shift register made of two flip-flops in a row.

PDKs and standard cells — the abstraction layer

The foundry supplies a **PDK (process design kit).** It contains the design rules (DRC rules), device models (SPICE parameters) and a standard-cell library (the layouts and timing characteristics of basic blocks such as NAND gates and flip-flops).

Why this is decisive: thanks to the PDK, a fabless company can design without knowing semiconductor physics. It plays exactly the role of an API/ABI in software. And as with an API, a design made for one PDK does not carry over to another foundry as it is. The standard cells and design rules change, so everything from synthesis through place and route, timing

closure and DRC/LVS has to be run again, and a new mask set has to be made. This porting cost is the real barrier to switching foundries, and it is where TSMC's ecosystem advantage comes from.

This ecosystem is a named institution. TSMC groups its partners into alliances under OIP (Open Innovation Platform®). When the Cloud Alliance was added as the fifth in October 2018, the lineup was EDA, IP, Design Center, VCA (then called Value Chain Aggregator, now Value Chain Alliance) and Cloud; in October 2022 the 3DFabric Alliance (advanced packaging and 3D IC) was formed as the sixth. Samsung Foundry does the same thing through its SAFE™ program. It bundles certified PDKs, design methodologies and IP, and runs IP, EDA, Cloud, Design Service, Multi-Die Integration (MDI) and OSAT alliances (checked October 2026). From the customer's side, choosing a foundry is not choosing one process but choosing this whole partner network.

---

4. The truth about "3nm" — how to read node names

What it is not

Node names once referred to a real physical dimension such as gate length. Since FinFET they have become **marketing names that correspond to no physical dimension at all.** Different companies' "3nm" processes differ in density and performance, and even across generations at the same company the name does not track the performance gain directly.

Then what should you look at?

METRIC	MEANING
Transistor density (MTr/mm²)	Millions of transistors per square millimeter. The most direct comparison metric, though it varies with the logic/SRAM mix
CPP (contacted poly pitch)	Spacing between gates
MMP (minimum metal pitch)	Minimum wiring pitch
Cell height (number of tracks)	The height of a standard cell counted in the horizontal routing tracks that run through it (in units of the minimum metal pitch). The T in 6T and 5T stands for track. It is a different word from the T in the 6T SRAM of Session 3 (six transistors)
PPA improvement	Gains in performance, power and area over the previous node. Always check the conditions behind vendor figures
SRAM cell area	**Especially important lately.** SRAM does not scale as well as logic, so chips with large caches gain little from a new node

> **What matters more for AI accelerators:** AI chip performance depends more on **memory bandwidth and packaging** than on the node. Attaching one more HBM stack can affect performance more than being one node generation ahead.

The 2nm class as of 2026

Status as rechecked on October 8, 2026 against each company's official announcements.

FOUNDRY	NODE	TRANSISTOR STRUCTURE	BACKSIDE POWER DELIVERY	STATUS
TSMC	N2	Nanosheet GAA	None (Super Power Rail introduced with A16. Production was planned for 2026 at the time of the April 2024 announcement; according to a TSMC executive's remarks reported from the April 2026 technology symposium, volume production is expected to begin in 2027. checked 2026-10-22)	Volume production began in Q4 2025. The enhanced N2P is scheduled for volume production in the second half of 2026
Intel	18A	RibbonFET (GAA)	**PowerVia (applied)**	First product, Core Ultra series 3, announced January 2026; first server product, Xeon 6+, launched in Q2 2026. The enhanced 18A-P entered risk production in June 2026
Samsung	SF2	MBCFET (GAA)	—	Volume production began in 2025. Used in the Exynos 2600

The contrast to watch: Intel chose the aggressive strategy of introducing GAA and backside power delivery at the same time; TSMC chose the conservative one of stabilizing GAA first and pushing backside power delivery to the next node. **This is not a difference in technical superiority but a difference in risk-budgeting strategy.** It makes a good discussion topic.

---

5. The economics of yield — this book's second key tool

The Poisson model

Assume that defects land on the wafer at random (spatially independent and uniform). If the average density of killer defects per unit area is D_0 [defects/cm^2] and the die area is A [cm^2], the number of defects landing on one die follows a Poisson distribution with mean $\lambda = AD_0$. The probability of no defect at all, that is, the yield, is

$$Y = P(X = 0) = e^{-AD_0}$$

For statistics majors: this is just the $P(X = 0)$ of a Poisson distribution. One of the pleasures of this field is that semiconductor yield theory starts from the first chapter of undergraduate probability.

Where the formula comes from — cutting the die into cells. Divide the die into n small cells of equal area, and let the probability of a defect landing in each cell be $p = \lambda / n$. If the cells are independent, the probability that the die is clean is the probability that every cell is clean, and as you cut the cells ever finer this becomes the high-school limit $\lim_{n \to \infty} (1 + x / n)^n = e^x$ with $x = -\lambda$.

$$P(0\text{ defects}) = \left(1 - \frac{\lambda}{n}\right)^n \longrightarrow e^{-\lambda} \quad (n \to \infty)$$

Calculating, for $\lambda = 0.8$ you get 0.4344 at $n = 10$, 0.4479 at $n = 100$ and 0.4492 at $n = 1{,}000$, while the limit $e^{-0.8}$ is 0.4493. Splitting into just ten cells already reaches 97% of the answer.

▾ Show

General k — the limit of the binomial. The probability that defects land in exactly k of the n cells is binomial. Write it as four factors.

$$P(X = k) = \binom{n}{k}\left(\frac{\lambda}{n}\right)^k\left(1 - \frac{\lambda}{n}\right)^{n-k} = \frac{\lambda^k}{k!} \cdot \frac{n(n-1)\cdots(n-k+1)}{n^k} \cdot \left(1 - \frac{\lambda}{n}\right)^n \cdot \left(1 - \frac{\lambda}{n}\right)^{-k}$$

Fix k and let $n \to \infty$; the last three factors go to 1, $e^{-\lambda}$ and 1. What remains is $P(X = k) = e^{-\lambda}\lambda^k / k!$, the Poisson distribution. This result, that the count of a rare event spread over very many opportunities follows a Poisson distribution, is called the law of rare events. With $k = 0$ the first, second and fourth factors all become 1, and only the third is left, which is the formula in the text.

Numerically, for $\lambda = 0.8$ and $k = 1$ the binomial probability is 0.3777 at $n = 10$, 0.3612 at $n = 100$ and 0.3596 at $n = 1{,}000$, and the Poisson value is 0.3595 (calculated).

Getting a feel for the numbers

With $D_0 = 0.1$ defects/cm² (an illustrative value at the level of a mature process):

DIE AREA	$\lambda = AD_0$	YIELD $e^{-\lambda}$
50 mm² (0.5 cm²)	0.05	95.1%
100 mm²	0.10	90.5%
200 mm²	0.20	81.9%
400 mm²	0.40	67.0%
600 mm²	0.60	54.9%
800 mm² (near the reticle limit)	0.80	**44.9%**

When area grows 16-fold, yield falls below half. And because a large die also fits fewer times on a wafer, the number of good dies falls much more steeply.

Good dies per wafer

The area of a 300mm wafer is about $\pi \times 15^2 = 706$ cm². There are several approximation formulas for gross die per wafer that account for edge loss; a common one is

$$\text{GDPW} \approx \frac{\pi d^2 / 4}{A} - \frac{\pi d}{\sqrt{2A}}$$

(d = wafer diameter, A = die area. The second term is the edge-loss correction: the wafer circumference divided by the die diagonal $\sqrt{2A}$. The comparison showing that this formula comes out 1–5% below the directly counted number for 100–800 mm² dies is in "A common sticking point" in section 0.)

Good dies = GDPW × Y, and **cost per die = wafer price / number of good dies.**

> **Exercise:** with the price of one wafer as W, find the ratio of cost per die for die areas of 100 mm² and 800 mm². The area ratio is 8, but the cost ratio comes out much larger.

Why the Poisson model is wrong — defect clustering

Real defects do not scatter independently; they **cluster** (particle contamination, tool faults, wafer-edge effects). With clustering, the Poisson model **underestimates** yield, because when defects crowd onto a few dies, the remaining dies stay clean.

Alternative models

MODEL	FORMULA	ASSUMPTION
Poisson	$Y = e^{-AD_0}$	Completely random
Murphy	$Y = \left(\frac{1 - e^{-AD_0}}{AD_0}\right)^2$	Defect density has a triangular distribution
Exponential (Seeds)	$Y = \frac{1}{1 + AD_0}$	Defect density has an exponential distribution
Negative binomial	$Y = \left(1 + \frac{AD_0}{\alpha}\right)^{-\alpha}$	**Defect density has a gamma distribution → reflects clustering**

In the negative binomial model, α is the clustering parameter: as $\alpha \to \infty$ it converges to Poisson, and the smaller α, the heavier the clustering. **That a gamma–Poisson mixture becomes a negative binomial is a familiar result in statistics, and the semiconductor yield model has exactly that structure.** If the defect density D follows a gamma distribution with mean D_0, its variance is D_0^2 / α. $\alpha = 1$ means the standard deviation of the defect density equals its mean (the integral is in the solution to Exercise 3). The memory yield model IBM published in 1980 was also a mixture model that shook the Poisson mean (Stapper et al., 1980).

> **Going deeper:** this section is where someone trained in statistics can most comfortably dig deeper. In real fabs, spatial statistics and machine learning are applied to wafer-map data (which dies at which positions failed) to classify defect patterns and trace them back to the tool that caused them. This connects to the inspection and metrology section of Session 5.

A is not the whole die area — critical area. Not every defect that lands causes a failure. A particle that falls between two wires does not short them unless it is big enough to touch both, and one that lands where there is no circuit does nothing. So strictly, the A in $\lambda = AD_0$ is not the die area but the **critical area**: the area of the places where a defect of that size would break the circuit. On the same die, the larger the defect, the larger the critical area, and densely wired blocks have more of it. IBM's memory yield model (Stapper et al., 1980)

described the product by this critical area and the process by the defect density, separately. When this book calls D_0 the "killer defect density" and uses the die area as it is, that is a simplification that absorbs the effect of critical area into the density.

What yield explains

1. **Why chiplets** — making four 200 mm² dies (82% each) separately, testing them, and binding only the good ones (KGD) is far better than one 800 mm² die (45% yield). Bind the four without sorting and the finished-product yield is $(e^{-0.2})^4 = e^{-0.8} \approx 45\%$, the same as the single die (the multiplicative yield of Session 3). The gain comes not from cutting the die smaller but from throwing out the bad ones before assembly so that only good silicon is used. The cost of joining them and the assembly yield are added, however. That profit-and-loss calculation is the core of Session 6.
2. **Why foundries sell by the wafer** — it is a question of who bears the yield risk. Under a per-wafer contract the fabless company bears it; under a per-good-die contract the foundry does. The newer the node, the more the foundry prefers per-wafer terms.
3. **Why new nodes are expensive** — early on, D_0 is high. Over time it comes down along the learning curve (yield learning). This is the structural advantage of the front-runner.
4. **Why binning is done** — binning comes in two kinds.
   - **Defect binning.** Some defects leave the chip usable if the affected core is disabled. Selling an 8-core design as a 6-core product is a strategy for recovering yield. Defects can also be repaired. Memory keeps spare rows and columns and swaps the defective lines for spares (repair), and chips that work only in part are sometimes sold separately. IBM, reporting the yield trend of its 64K-bit RAM, also presented the yield gains it obtained from redundant circuits and partially good products (Stapper et al., 1982). The redundant circuits of HBM in Session 3 work on the same principle.
   - **Speed and power binning.** Even among defect-free dies, process variation makes the maximum operating frequency and power differ. So each die is tested for its maximum frequency and graded. This is a matter of **parametric yield,** which e^{-AD_0}, a formula that counts defects, cannot explain. Where defect binning deals with "how many are alive," speed binning deals with "how fast the living ones are."

---

6. ASIC, FPGA or general-purpose processor — a quantitative choice

Break-even (a refinement of the formula from Session 2)

$$N^* = \frac{C_{\mathrm{NRE}}}{(c_G + o_G) - (c_A + o_A)}$$

- c: unit acquisition cost; o: present value of lifetime operating cost (mainly electricity)
- As in Session 2, assume one ASIC replaces the work of one GPU 1:1. If throughput differs, N has to be converted into the number of units that handle the same demand
- c_A is calculated from the cost per die in section 5 above + packaging + test

Trade-offs on five axes

	DEVELOPMENT TIME	NRE	UNIT COST	POWER EFFICIENCY	FLEXIBILITY
General-purpose CPU/GPU	0 (purchase)	0	High	Low	Highest
FPGA	Months	Low	Very high	Low to medium	Reconfigurable
Structured ASIC	About 1 year	Medium	Medium	Medium	Limited
Full-custom ASIC	2–3 years	Very high	Low	Highest	None

The trap on the time axis

Even with enough volume to clear N^*, it is no use if the period over which that chip handles that volume is short. If model architectures change substantially every two years and chip development takes two to three years, the computation pattern the chip targeted must still dominate when it is finished.

Why hyperscalers can take this risk

1. Their volume N is overwhelming.
2. **They control their own workloads.** Being able to design the model to fit the chip is decisive.
3. The total savings on electricity are large (at data center scale). Per unit, it is mainly the difference in acquisition price that decides the break-even count, and the difference in electricity cost amounts to only a few percent to a dozen-odd percent (solution to Exercise 4 in Session 2, solution to Exercise 5 in this session); but run tens of thousands of units for several years and that share also becomes a large sum.
4. There is option value in bargaining power with vendors.
5. They have the capital to absorb a failure.

Why most companies do not: none of the five holds for them.

---

7. Common misconceptions

Misconception 1: "TSMC also designs chips" → The core of the pure-play foundry model is the trust that it does not compete with its customers. It offers IP blocks and packaging services but does not design finished chips. This neutrality is a large part of TSMC's success.

Misconception 2: "A more advanced node makes a faster chip" → For AI accelerators, memory and packaging can matter more. And the node name itself is not a comparable metric.

Misconception 3: "Yield converges to 100% as a process matures" → Inside the Poisson formula, that is right. Whatever the area, $e^{-AD_0} \to 1$ as $D_0 \to 0$. The trouble is that D_0 never goes all the way to 0. At the same D_0, yield falls exponentially with die size, so even on a

mature process a large die stays at a lower yield than a small one (an 800 mm² die is at 67.0% even with $D_0 = 0.05$). On top of that, parametric losses that are not counted as defects (speed and power binning in section 5) remain outside the formula. And a new node always starts again from low yield.

Misconception 4: "A fabless company doesn't need to know manufacturing" → The PDK abstracts it away, but high-performance design has to take process characteristics (variability, electromigration, heat) deeply into account. That is why DFM (design for manufacturability) exists as a field of its own.

Misconception 5: "An OSAT is a low-value-added subcontractor" → Its standing has risen sharply now that advanced packaging decides performance. The leading edge, however, is being absorbed by foundries, so the OSAT's position is in flux.

Misconception 6: "EDA is just CAD software" → Place and route and timing optimization are large-scale combinatorial optimization problems, and reinforcement learning and graph neural networks are now actually being applied to them. **This is the most direct point of entry for people trained in CS and AI.**

---

8. Quantitative exercises

Exercise 1 — Yield curves For each of $D_0 = 0.05, 0.1, 0.2$, plot the yield curve over die areas of 50–800 mm². From what area does yield drop sharply?

Exercise 2 — Cost per die With the price of a 300mm wafer as W, find the cost per die for die areas of 100 / 400 / 800 mm² as multiples of W. Apply both the GDPW approximation and the yield.

Exercise 3 — Poisson vs. negative binomial For the same AD_0, compare the yields of the Poisson model and the negative binomial ($\alpha = 1, 2, 5$). Confirm that the heavier the clustering, the more the actual yield exceeds the Poisson prediction, and explain why in probabilistic terms.

Exercise 4 — Chiplet profit and loss (preview of Session 6) Compare a single 800 mm² die with four 200 mm² chiplets. On the chiplet side, include (a) a 10% area overhead for the interface circuits between chiplets, (b) an assembly yield of 95% and (c) the added cost of the interposer and assembly. Take (c) as the cost loaded onto one finished product (including the share of failed assembly attempts). Which side wins? Is there a point where changing D_0 reverses the conclusion?

Exercise 5 — The ASIC decision Assumptions: NRE USD 150 million, ASIC unit cost USD 4,000, commercial GPU unit cost USD 30,000, three-year electricity cost USD 2,000 for the ASIC and USD 6,000 for the GPU. Find N^*, and discuss which organizations could realistically reach that volume.

9. Discussion questions

1. Can the foundry–fabless division of labor be reversed? Which direction does each of Intel's IDM 2.0 strategy and hyperscalers' vertical integration point in?
2. Between the risk-budgeting strategies of Intel (GAA and backside power at once) and TSMC (GAA first, backside power later), which is rational? Under what conditions is each right?
3. Yield data are trade secrets. What observable proxies could be used to estimate yield from outside? (Prices, lead times, product lineups, binning structure, etc.)
4. Why does the three-firm EDA oligopoly not break? Could open-source EDA (OpenROAD and others) and open PDKs change that?
5. Now that node names have lost their physical meaning, why does the industry keep using that naming scheme? Is there an alternative?

10. Further study

- **Standard VLSI design textbook:** Weste & Harris, *CMOS VLSI Design: A Circuits and Systems Perspective* — sets out models of yield, area and power
- **Yield theory:** Stapper's papers on the IBM yield model — Stapper, McLaren, Dreckmann, "Yield Model for Productivity Optimization of VLSI Memory Chips with Redundancy and Partially Good Product," *IBM J. Res. Dev.* 24(3):398–409 (1980), doi:10.1147/rd.243.0398 · Stapper et al., "Evolution and Accomplishments of VLSI Yield Management at IBM," *IBM J. Res. Dev.* 26(5):532–545 (1982), doi:10.1147/rd.265.0532. For a comparison of several yield models, Cunningham, "The Use and Evaluation of Yield Models in Integrated Circuit Manufacturing," *IEEE Trans. Semiconductor Manufacturing* 3(2):60–71 (1990), doi:10.1109/66.53188
- **GDPW approximations:** de Vries, "Investigation of Gross Die Per Wafer Formulas," *IEEE Trans. Semiconductor Manufacturing* 18(1):136–139 (2005), doi:10.1109/TSM.2004.836656 — worth reading after you have worked out the table in section 0 yourself
- **Try it yourself:** Tiny Tapeout — a project that lets you put your own design on a real chip for a small fee. **The best way for someone with a CS background to feel the tape-out pipeline firsthand**
- **Open toolchain:** OpenLane / OpenROAD + SkyWater 130nm open PDK — lets you run from RTL to GDS for free
- **AI × EDA:** Mirhoseini et al., "A graph placement methodology for fast chip design," *Nature* 594 (2021), doi:10.1038/s41586-021-03544-w — placement based on reinforcement learning. The paper's reproducibility was disputed, so read the later rebuttals and replies along with it

- **Industry structure:** IR materials and technology-symposium presentations from TSMC, Intel and Samsung — TSMC quarterly results · Intel press releases

---

Self-study appendix A — solutions to the quantitative exercises

Solutions to the exercises in section 8. All areas are converted to cm² before being plugged in (100 mm² = 1 cm²).

Solution to Exercise 1 — Yield curves

▾ Solution

$Y = e^{-AD_0}$.

DIE AREA	$D_0 = 0.05$	$D_0 = 0.1$	$D_0 = 0.2$
50 mm²	97.5%	95.1%	90.5%
100 mm²	95.1%	90.5%	81.9%
200 mm²	90.5%	81.9%	67.0%
400 mm²	81.9%	67.0%	44.9%
600 mm²	74.1%	54.9%	30.1%
800 mm²	67.0%	**44.9%**	**20.2%**

How to answer "from what area does it drop sharply"

An exponential has no separate knee. Fix a reference line instead, and a clean formula comes out. **The area at which yield is 50%** is

$$Y = 0.5 \;\Rightarrow\; A_{50} = \frac{\ln 2}{D_0} = \frac{0.693}{D_0}\ \text{cm}^2$$

D_0	A_{50}
0.05	1,386 mm² (larger than the reticle limit — by the 50% criterion, the exposure area blocks you first. Even at the reticle limit of 858 mm², yield is 65.1%, so the effect on cost remains)
0.1	**693 mm²**
0.2	**347 mm²**

How to read it: when D_0 doubles, the die area allowed at the same target yield (here 50%) **halves.** 50% is only a reference line chosen for the explanation; chips larger than this area can still be made. Their cost simply rises as yield falls. A high D_0 early in a new node means it is hard to make large chips at an affordable cost, and this is both the structural advantage of the leading foundry and one of the reasons large AI accelerators cannot move to the newest node right away. The largest size that can

actually be made is set not by yield but by the reticle limit.

Solution to Exercise 2 — Cost per die

▾ Solution

Set $d = 30$ cm and $D_0 = 0.1$, and apply both GDPW and yield.

$$\text{GDPW} \approx \frac{\pi \cdot 900 \,/\, 4}{A} - \frac{\pi \cdot 30}{\sqrt{2A}} = \frac{706.9}{A} - \frac{94.25}{\sqrt{2A}}$$

DIE AREA	GDPW	YIELD	GOOD DIES	COST PER DIE
100 mm² (1 cm²)	640.2	0.905	579.3	$0.00173\,W$
400 mm² (4 cm²)	143.4	0.670	96.1	$0.01040\,W$
800 mm² (8 cm²)	64.8	0.449	29.1	$0.03435\,W$

Worked example (800 mm²): $706.9 / 8 = 88.36$, $94.25 / \sqrt{16} = 23.56$, GDPW $= 64.80$. Good dies $= 64.80 \times 0.4493 = 29.1$. Cost $= W / 29.1$.

Key result

$$\frac{0.03435}{0.00173} \approx 19.9\times$$

The area is 8 times larger, but the cost is 20 times larger. It is the product of three things — 8 times from area, 2.0 times from yield and 1.2 times from edge loss.

Implications

- The silicon cost of one 800 mm²-class AI accelerator equals that of twenty 100 mm²-class chips. The accurate statement is **not "big chips are expensive" but "chips get superlinearly more expensive with area."**
- This superlinearity is **the economic driver of chiplets.** If the same total area can be split into small pieces, cost drops sharply. Exercise 4 calculates this.
- Here you can actually fill in the c_A of Exercise 4 in Session 2. Put one wafer at USD 15,000 and the silicon cost alone of an 800 mm² die is about USD 515, with HBM, packaging and test on top.

The figure below shows a die area of 800 mm², defect density $D_0 = 0.1$, and a layout with a die corner aligned to the wafer center. In the screen edition you can change the die area, grid position, D_0, wafer price and defect model and watch how the directly counted number, the two approximation formulas, the yield and the cost per die move. The defect dots in the figure are one random sample; the yield and cost in the readout are calculated from the formulas.

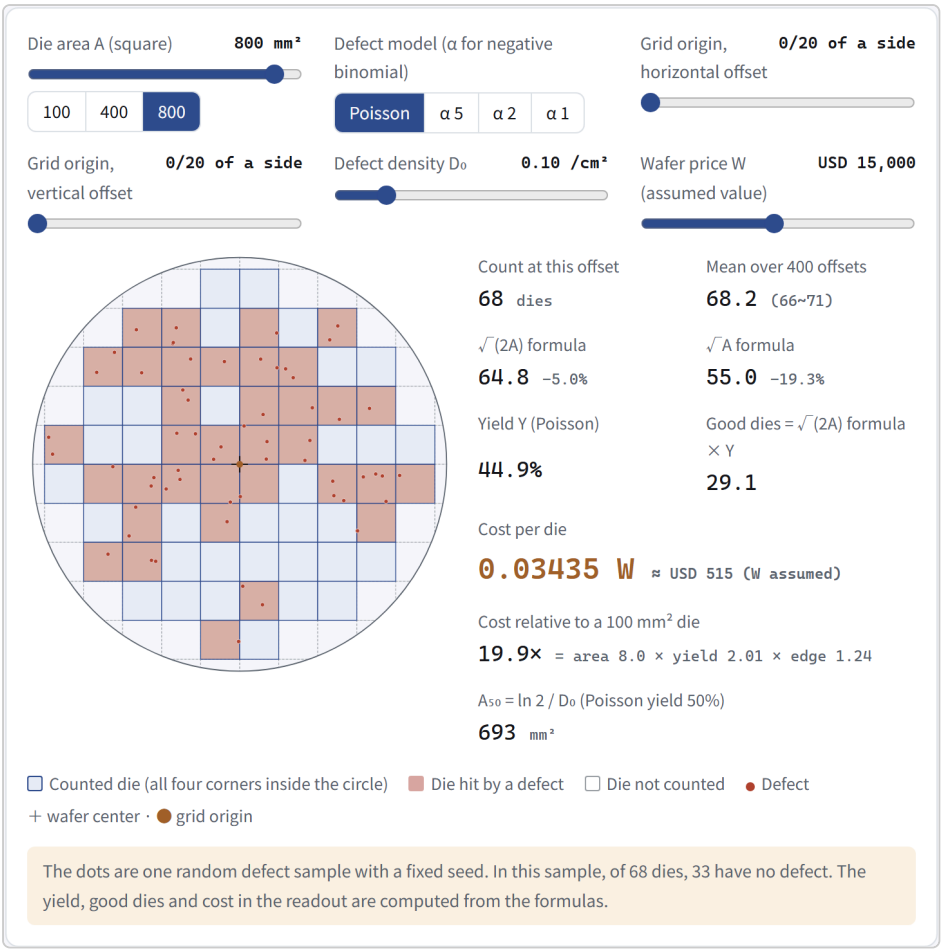


Figure 4-1. Laying 800 mm² dies on a 300 mm wafer gives a direct count of 68 (66–71 depending on grid position); the √(2A) approximation gives 64.8. With $D_0 = 0.1$, yield is 44.9%, good dies number 29.1, and cost per die is 0.03435 times the wafer price, about 19.9 times that of a 100 mm² die. The defect dots are one random sample. — In the HTML edition you can change the values here (HTML edition, Session 4, Solution to Exercise 2).

Solution to Exercise 3 — Poisson vs. negative binomial

▾ Solution

Fix $\lambda = AD_0 = 0.8$ (800 mm², $D_0 = 0.1$) and compare.

MODEL	FORMULA	YIELD
Poisson	$e^{-0.8}$	44.93%
Negative binomial $\alpha = 5$	$(1 + 0.16)^{-5}$	47.61%
Negative binomial $\alpha = 2$	$(1 + 0.40)^{-2}$	51.02%
Negative binomial $\alpha = 1$	$(1 + 0.80)^{-1}$	**55.56%**

The heavier the clustering (the smaller α), the higher the yield, even though the total expected number of defects is exactly the same.

The probabilistic explanation — one line of Jensen's inequality

Suppose the defect density itself varies from wafer to wafer. That is, Λ is a random variable with $E[\Lambda] = \lambda$. The yield is the average of the conditional Poisson $P(X = 0)$, so

$$Y = E\left[e^{-\Lambda}\right] \geq e^{-E[\Lambda]} = e^{-\lambda}$$

Because e^{-x} is a **convex function,** Jensen's inequality always holds in this direction. **The more defects cluster (the larger the variance), the higher the yield must be relative to the Poisson prediction.**

Set $\Lambda \sim \text{Gamma}(\alpha, \lambda / \alpha)$ and the expectation above is exactly the moment-generating function of the gamma distribution, whose value is $(1 + \lambda / \alpha)^{-\alpha}$ — the negative binomial yield formula. As $\alpha \to \infty$ the variance goes to 0 and it converges to Poisson.

Writing out the integral. Let the defect density D follow a gamma distribution with mean D_0 and shape parameter α, and average the conditional yield e^{-AD} over its density.

$$Y = \int_0^\infty e^{-AD} \frac{(\alpha / D_0)^\alpha}{\Gamma(\alpha)} D^{\alpha - 1} e^{-\alpha D / D_0} \, dD = \frac{(\alpha / D_0)^\alpha}{(A + \alpha / D_0)^\alpha} = \left(1 + \frac{AD_0}{\alpha}\right)^{-\alpha}$$

The middle step plugs $c = A + \alpha / D_0$ into the definition of the gamma function, $\int_0^\infty D^{\alpha - 1} e^{-cD} \, dD = \Gamma(\alpha) / c^\alpha$. The variance of this gamma distribution is as follows.

$$\text{Var}\,(D) = \frac{D_0^2}{\alpha}$$

With this formula you can read α as a number rather than under the name "clustering parameter." At $\alpha = 1$ the standard deviation of the defect density equals its mean; at $\alpha = 5$ it is about 45% of the mean $(1 / \sqrt{5})$.

Checking numerically (calculated): at $D_0 = 0.1$ and $A = 8$, the numerical value of the integral above and the closed form agree at 0.555556, 0.510204 and 0.476113 for $\alpha = 1, 2, 5$, and the variances are 0.01, 0.005 and 0.002, equal to D_0^2 / α. These are the negative binomial yields in the table above.

Intuition: if 100 defects are spread evenly over the whole wafer, almost every die gets hit once. If the same 100 crowd onto 10 dies, all the rest are clean. **Yield is a question not of "how many defects there are" but of "how many dies are contaminated."**

Practical implication: using Poisson estimates yield conservatively (low). That is why fabs estimate α from actual wafer-map data, and the classification of those wafer-map patterns is where machine learning comes in, in the inspection and metrology section of Session 5.

The figure below shows a die split into 10 cells at $\lambda = 0.8$, and a comparison of Poisson and negative binomial ($\alpha = 2$) for an 800 mm² die at $D_0 = 0.1$. In the screen edition you can change the number of cells, λ, the die area, D_0 and α. The two small grids on the

right are a sample with the same number of defects scattered evenly and a sample scattered according to a density shaken by a gamma.



Figure 4-2. At $\lambda = 0.8$, splitting the die into 10 cells gives $(1 - \lambda/n)^n = 0.4344$, 97% of the limit $\exp(-0.8) = 0.4493$. At the same λ, the negative binomial yield ($\alpha = 2$) is 51.0%, higher than the Poisson 44.9%, and the variance of defect density is $D_0^2/\alpha = 0.005$. — In the HTML edition you can change the values here (HTML edition, Session 4, Solution to Exercise 3).

Solution to Exercise 4 — Chiplet profit and loss (preview of Session 6)

▾ Solution

A single 800 mm² die vs. four 200 mm² chiplets. $D_0 = 0.1$, interface overhead 10%, assembly yield 95%.

Single die: 0.03435 W from Exercise 2 above

Chiplets: 10% overhead on each 200 mm² → 220 mm² = 2.2 cm²

- GDPW $= 706.9 / 2.2 - 94.25 / \sqrt{4.4} = 321.3 - 44.93 = 276.4$
- Yield $= e^{-0.22} = 0.8025 \rightarrow$ good dies $= 221.8$
- Per chiplet $W / 221.8 = 0.004509\,W$; for four, $0.018035\,W$ (the KGD assumption that only good chiplets are used. Dividing by the number of good dies to get the cost

means exactly that)

- Dividing by the assembly yield of 0.95 gives $0.018984\,W$

Savings $= 0.03435 - 0.018984 = 0.01536\,W$

Verdict: the chiplets win if the added interposer and assembly cost per finished product is **less than about 1.5% of the price of a wafer (0.01536 W).** With a wafer at USD 15,000, that means a margin of about USD 230 per system.

If the added cost is incurred at every assembly attempt, the share of failed attempts is also divided by 0.95, and the cost of a finished product becomes $(4c + C)\,/\,0.95$. The threshold is then $C < 0.95 \times 0.03435 - 0.018035 = 0.01459\,W$, about USD 219 per attempt, slightly lower. Whichever basis you use, the direction of the conclusion is the same.

Where the conclusion flips — changing D_0

D_0	SINGLE	FOUR CHIPLETS	SAVINGS	VERDICT
0.02	$0.01811\,W$	$0.01592\,W$	$0.0022\,W$	Reverses once assembly cost exceeds just **0.22%** of the wafer price
0.1	$0.03435\,W$	$0.01898\,W$	$0.0154\,W$	Chiplets ahead

What to read from this example: the cost advantage of chiplets is larger the worse the yield. Under the conditions of the same wafer price, the same D_0, a four-way split, 10% overhead and 95% assembly yield, the advantage shrinks quickly as the process matures and D_0 falls. This fits the fact that chiplets were adopted first for the newest nodes and large-area chips. But the reasons for using chiplets also include area beyond the reticle limit, heterogeneous integration that uses a different process for each block, and design reuse (Session 6), so this one calculation cannot settle where chiplets are and are not used.

> In Exercise 1 of Session 6, the number of pieces n becomes a free variable and the optimum is found. To give away the result — **the optimal number of pieces shifts with D_0.**

The same calculation can be done with this exercise's assumptions in the figure at the end of the solution to Exercise 1 in Session 6 (screen edition).

Solution to Exercise 5 — The ASIC decision

▾ Solution

NRE USD 150 million, ASIC USD 4,000, GPU USD 30,000, three-year electricity cost USD 2,000 and USD 6,000 respectively.

$$N^* = \frac{1.5 \times 10^8}{(30{,}000 + 6{,}000) - (4{,}000 + 2{,}000)} = \frac{1.5 \times 10^8}{30{,}000} = 5{,}000 \text{ units}$$

Organizations for which this volume is realistic

TYPE OF ORGANIZATION	UNITS DEPLOYED OVER 3 YEARS	VERDICT
Hyperscaler	Hundreds of thousands	Far above — an in-house chip is rational
Large AI lab	Tens of thousands	Above — but the risk of workload change is large
In-house AI infrastructure at a large company	Hundreds to thousands	**Borderline.** Usually below
Ordinary companies, research institutes	Dozens	Clearly below

The structural implication for the industry

A threshold of 5,000 units is lower than you might expect. So **operators that are "almost hyperscaler-class" can consider custom silicon** even if they are not hyperscalers, and the businesses that take on this demand are custom-ASIC design services such as Broadcom and Marvell. The structural reason for their recent rapid rise lies in this calculation.

But the warning from Exercise 4 in Session 2 applies in full. N^* is only a necessary condition. The sufficient condition is that **the workload still be alive two to three years later, after development,** and the only organizations that can guarantee this are those that control their own models.

The figure below uses the assumed values of Exercise 5 (NRE USD 150 million, ASIC USD 4,000, GPU USD 30,000, three-year electricity USD 2,000 / USD 6,000). In the screen edition you can change the five values and see which organization size in the table above N^* falls into. Section 6 defined o as the present value of operating cost but did not fix a discount rate, so the figure uses the three-year electricity cost undiscounted. The boundaries of the organization bands translate the table's "dozens, hundreds to thousands, tens of thousands, hundreds of thousands of units" into orders of magnitude.

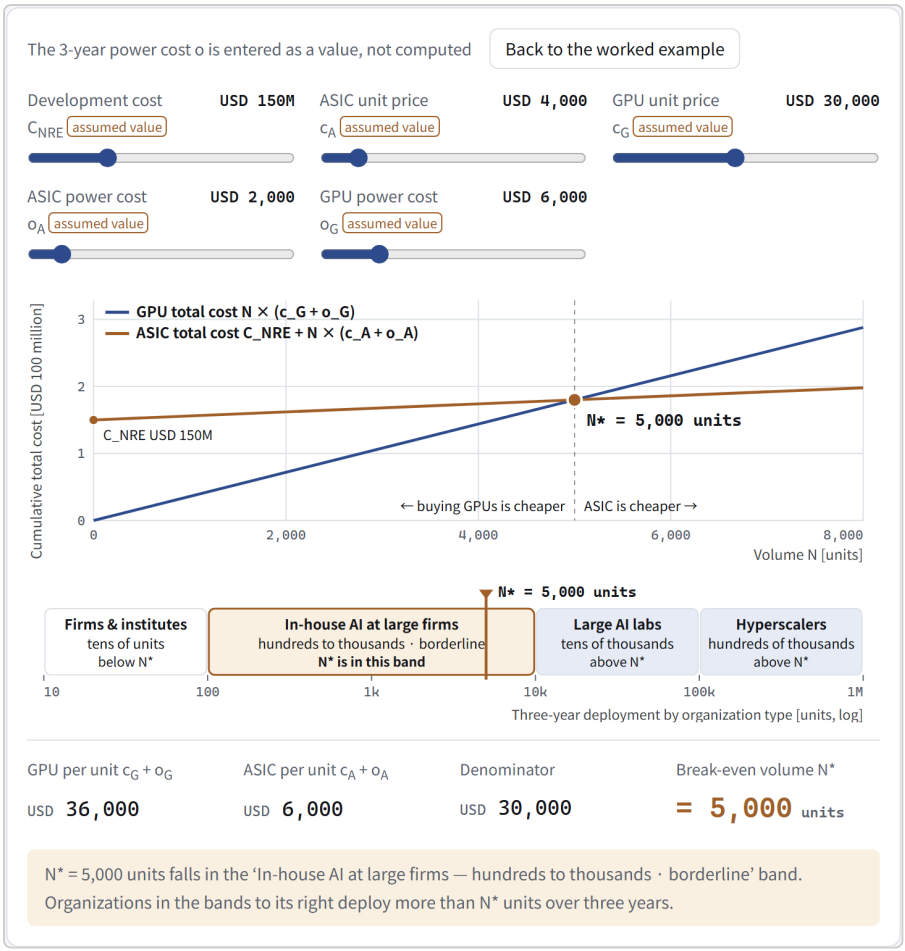


Figure 4-3. With the assumptions of Exercise 5 (NRE USD 150 million, ASIC USD 4,000 and GPU USD 30,000, 3-year power cost USD 2,000 and 6,000), N* = 5,000 units, which falls inside the band "in-house AI infrastructure at a large company: hundreds to thousands of units." — In the HTML edition you can change the values here (HTML edition, Session 4, Solution to Exercise 5).

Self-study appendix B — leads for the discussion questions

QUESTION	KEY AXIS	LEADS
1. Can the division of labor be reversed?	**Capital intensity**	The division happened because fab investment exceeded the demand of any single company. Reversing it would require one company's demand to be large enough to fill a fab — and **hyperscalers' AI demand is actually approaching that scale.** This is the basis for the return of vertical integration, and its logic runs in a different direction from Intel's IDM 2.0
2. Risk budgeting for GAA and backside power	**Serial vs. parallel risk**	What can be observed: introduce two changes at once, and when something goes wrong it is hard to tell which one caused it, and a delay in either one pushes back the whole schedule. Doing them in sequence keeps that risk small but takes more time. A hypothesis for discussion: the one behind has an incentive to go simultaneous to catch up, and the one ahead has an incentive to go sequential to hold its place. Whether the hypothesis is right has to be weighed against the probability of success of each choice, the losses a delay causes and the reward for the leading position, and those values are not set here
3. Observable proxies for yield	**Prices carry information**	Lead times, binning structure (a large share of lower SKUs signals low yield), the speed of price cuts, early volumes of new products, the form of the foundry contract (per wafer or per good die — see section 5). **The form of the contract is the most honest signal**
4. Why the EDA oligopoly does not break	**The value of what has been proven**	The cost of switching tools is not the price of the tool but **the cost of re-verification.** Would you print a mask set worth tens of millions of dollars on the strength of a new tool's verification results alone? Open-source EDA cannot skip this accumulation of trust
5. Why keep using node names?	**The name as a coordination device**	Though they have no physical meaning, they still work as **a common coordinate for naming generations.** The alternatives (density metrics, performance and power metrics) are more accurate but do not compress into a single number, which makes them hard to use in marketing and contract negotiations

Self-study appendix C — key terms

TERM	ONE-LINE DEFINITION
Defect density D_0	Number of killer defects per unit area [defects/cm^2]
Critical area	The area of the places where a defect of a given size would break the circuit. Larger for larger defects
Die	One chip cut from a wafer
GDPW (gross die per wafer)	The total number of dies from one wafer (before yield is applied)
Good die	A die that works correctly. GDPW × yield
Poisson yield model	$Y = e^{-AD_0}$. Assumes defects are completely random
Negative binomial (yield model)	$Y = (1 + AD_0 \,/\, \alpha)^{-\alpha}$. Reflects clustering

TERM	ONE-LINE DEFINITION
Clustering parameter α	The smaller it is, the more defects cluster. Poisson at ∞
Yield learning	The process by which D_0 comes down over time
Binning	Sorting dies into grades for sale. There is defect binning, which disables bad cores and sells the die as a lower product, and speed and power binning, which grades dies by maximum frequency and power
Chiplet	Making a large chip as several small dies and joining them
Reticle limit	The largest area a lithography tool exposes in one shot (about 26×33 mm)
NRE (non-recurring engineering)	One-time cost of development
Structured ASIC	An intermediate form that customizes only some layers to lower NRE
IDM (integrated device manufacturer)	A semiconductor company that both designs and manufactures
PDK (process design kit)	The bundle of process-specific design rules and models a foundry supplies
Track	The slot for one horizontal wire running through a standard cell. Cell height is counted in tracks (6T and 5T cells). Not the same T (transistor) as in 6T SRAM

Self-study appendix D — self-assessment

☐ I can write $Y = e^{-AD_0}$ and explain why it is an exponential (the Poisson $P(X = 0)$) → section 5 ★

☐ I can **break down into three factors** why cost grows 20-fold when area grows 8-fold → Exercise 2 ★

☐ I can derive $A_{50} = \ln 2 \,/\, D_0$ and say that when D_0 doubles, the die area allowed at the same target yield halves → Exercise 1

☐ I can explain what each of the two terms of the GDPW formula is → self-study guide

☐ I can explain that clustering **raises** yield, and why (Jensen's inequality) → Exercise 3

☐ I can show by calculation that the cost advantage of chiplets grows **the higher D_0 is** and shrinks as it falls → Exercise 4 ★

☐ I can calculate N^* and say why it is only a necessary condition → Exercise 5

☐ I can explain what the form of a foundry contract (per wafer vs. per good die) allocates → section 5

Before the next session

This is as far as **manufacturing seen from the outside** goes. From Session 5 we actually go inside.

QUESTION LEFT BY THIS SESSION	ANSWERED IN LATER SESSIONS
Where does D_0 come from?	**Session 5** — defects from each of roughly 1,000 process steps
Why does every process step have to be so precise?	**Session 5, Exercise 4** — the exponential accumulation of step yields
Why does the reticle limit set chip size?	**Session 5, Exercise 5** → **Session 6** chiplets
How are chiplets actually joined?	**Session 6** — interposers, TSVs, hybrid bonding

---

Sources

Introductory material chosen by the study curriculum (the "fabless" item)

CURRICULUM SUBTOPIC	LINK	HOW THIS BOOK USES IT
Terminology	Samsung Semiconductor Newsroom, "Foundry? Fabless? The Semiconductor Ecosystem at a Glance!" (in Korean)	An introduction to the division of labor. The basis of section 2. **Good material for opening a presentation**
Fabless company trends	ScienceON JAKO202470953293995, "Trends in Supporting Technologies for Advanced AI Semiconductors and Compilers" (in Korean)	A paper surveying AI chip designers in Korea and abroad (Sapeon, Rebellions, FuriosaAI; Google, Meta, Tesla) and compilers (MLIR, TVM, XLA). Korean cases for the fabless layer in section 2
Fabless company trends	ScienceON JAKO202369265025811, "Technical Trends in Hyperscale Artificial Intelligence Processors" (in Korean)	A report on the technology trends of companies in Korea and abroad developing processors for hyperscale AI (NVIDIA, Graphcore, Tesla, Google, Meta and three Korean firms). Background to the cases for the fabless layer in section 2
ASIC	ScienceON ASTI00673636, "Market Insight 336: AI Application-Specific Semiconductors (AI ASIC)" (in Korean) (report)	A technology and industry report on AI ASICs. Background to section 6
Overall process	Winspec, "What Kinds of Semiconductor Processes Are There? Front End and Back End Compared" (in Korean)	An overview of the whole semiconductor process. **Suitable preparatory reading for the pipeline in section 3 and the front end in Session 5**

Primary sources added in this revision (checked 2026-10-08)

- TSMC, Open Innovation Platform® · TSMC, Value Chain Alliance · TSMC 3DFabric® Alliance — design houses in section 2, OIP in section 3
- TSMC press release (2018-10-03), "TSMC Announces 5th Open Innovation Platform® Alliance, the OIP Cloud Alliance" · TSMC press release (2022-10-27), "TSMC Launches OIP 3DFabric Alliance ..." — alliance lineup and years in section 3
- Samsung Foundry, SAFE™ program — the Design Service Alliance (DSP, VDP) in section 2, the makeup of SAFE in section 3
- SEMI P39, Specification for OASIS® – Open Artwork System Interchange Standard — layout file format in section 3
- IEEE Std 1364-2005 (Verilog HDL) — nonblocking assignment in section 3 (superseding standard IEEE Std 1800)
- TSMC 2nm (N2) technology page · TSMC press release (2024-04-24), A16 announcement · Tom's Hardware (2026-04-22), technology symposium report: A16 volume production in 2027 (quoting a TSMC executive) — table in section 4
- Intel Q4 and full-year 2025 results press release (2026-01-22) · Intel Q2 2026 results press release (2026-07-23) · Intel Foundry VLSI Symposium press release (2026-06-16) — table in section 4
- Samsung Foundry logic node page · Exynos 2600 product page — table in section 4
- SK hynix Newsroom (2024-04-19), "SK hynix Partners with TSMC to Strengthen HBM Technological Leadership" — HBM4 base die in section 2
- Stapper, McLaren, Dreckmann (1980), doi:10.1147/rd.243.0398 · Stapper, Castrucci, Maeder, Rowe, Verhelst (1982), doi:10.1147/rd.265.0532 — critical area, mixture model, repair and partially good products in section 5 (bibliographic data from Crossref, content checked against the publisher's abstracts)
- Zhang, Ren, Su, "On-Chip Structures for Fmax Binning and Optimization," *Sensors* 22(4):1382 (2022), PMC8963108 · Datta et al., "Speed binning aware design methodology to improve profit under parameter variations," ASP-DAC 2006, pp. 712–717 (bibliographic data only, checked through Crossref) — speed binning in section 5
- de Vries (2005), doi:10.1109/TSM.2004.836656 — literature comparing GDPW approximations in section 0 (bibliographic data only checked; full text not read)

Verification status

- Status of the 2nm-class nodes (table in section 4) — rechecked against each company's official pages and press releases on 2026-10-08. TSMC N2 volume production began in Q4 2025 and N2P is scheduled for volume production in the second half of 2026 (TSMC 2nm technology page; the N2 wording "TSMC's 2nm (N2) technology has started volume production in 4Q25 as planned" rechecked 2026-10-22); Intel 18A's first product, Core Ultra series 3, announced (2026-01-22 results press release), Xeon 6+ launched (Q2 2026 results press release), 18A-P risk production (2026-06-16); Samsung SF2 "Started mass production in 2025" (logic node page) and Exynos 2600 "Based on the industry's first 2nm GAA process" (product page; both wordings rechecked 2026-10-22). Transistor structure names (nanosheet, RibbonFET, MBCFET) and the application of PowerVia carry over from the 2026-08-02 check
- TSMC A16's Super Power Rail (backside power delivery) and its production schedule — the 2024-04-24 press release (production planned for 2026) checked in the original (2026-10-08). The expected start of volume production in 2027 is confirmed only by a report (Tom's Hardware, 2026-04-22) of a TSMC executive's remarks at the 2026-04-22 technology symposium; TSMC's press release of the same day does not mention A16. TSMC's third-quarter earnings press release of 2026-10-15 does not mention A16 either, and there is still no official announcement that volume production has begun (checked 2026-10-22)
- TSMC's five OIP alliances (2018) and the 3DFabric Alliance (October 2022, the sixth) — checked against the original TSMC press releases (2026-10-08)
- The makeup of Samsung's SAFE™ (PDK, design methodology and IP certification; the DSP and VDP of the Design Service Alliance) — checked against Samsung's official page (2026-10-08)
- The nature of design houses and TSMC's VCA — confirmed on TSMC's official page (2026-10-08). The member list changes over time and is not given in the text
- SK hynix–TSMC memorandum of understanding on HBM4 collaboration (2024-04-19) — checked against the SK hynix newsroom original (2026-10-08; quoted wording rechecked 2026-10-22)
- Formulas of the Poisson, Murphy, Seeds and negative binomial yield models — the standard formulation of semiconductor yield theory
- The binomial-to-Poisson limit values, the gamma-mixture integral and $\mathrm{Var}\,(D) = D_0^2 \,/\, \alpha$ — recalculated in Python (2026-10-08)
- The meaning of the second term of the GDPW approximation (die diagonal $\sqrt{2A}$) and the comparison with directly counted numbers — calculated directly by laying a die grid at 400 positions over a 300 mm wafer (2026-10-08)
- Critical area, the mixture structure of IBM's memory yield model, and the yield gains from repair and partially good products in the 64K-bit RAM — checked against the publisher's abstracts of Stapper et al. (1980, 1982) (2026-10-08). The full texts could not be read, so no figures or table values from them are carried over
- That speed binning arises from process variation — checked against the original of Zhang et al. (2022) (2026-10-08)

- That OASIS is the layout exchange format succeeding the GDSII stream — checked on the SEMI P39 specification page (2026-10-08)
- The meaning of nonblocking assignment — only the IEEE Std 1364-2005 standard page was checked; the text of the standard could not be read. The meaning itself is standard Verilog behavior
- 300mm wafer specification — standard

Unconfirmed or reserved (not asserted in the text)

- **Fab construction cost, mask-set cost, scale of NRE** — vary widely by node, company and contract, and are rarely disclosed officially. The text mentions them only by order of magnitude
- **Defect density D_0 values** — company trade secrets. The 0.05–0.2 in the text are **assumed values for worked examples,** not the actual values of any particular company
- **Yield by company and node** — press reports (for example, SF2 yields in the 30–60% range) often have unclear sources, so they are not given as figures in the text
- **Specific transistor-density figures (MTr/mm^2)** — measurement bases differ by vendor, which makes direct comparison hard. Only the existence of the metric and how to read it are described
- **Unit prices, NRE and electricity costs in the exercises of section 8** — all assumed values for illustration, not actual market prices

SESSION 5

Front-End Processing

KEY TAKEAWAY

The eight major front-end processes are not a list to memorize but a single repeating loop: "draw a pattern → add or remove material → planarize → inspect." Repeated layer after layer, this loop becomes hundreds of steps. Lithography sits at its center, and lithography comes down to one equation, the Rayleigh criterion $\mathrm{CD} = k_1 \lambda$ / NA. Shorten the wavelength, enlarge the numerical aperture, or shrink the process factor. **The whole history of semiconductor scaling is the history of attacking these three terms.** Why transistors went from planar to FinFET to GAA also fits in one sentence: as channels grew shorter, the gate lost its grip, and the only remedy was to increase the area over which the gate wraps the channel.

PRIMARY SOURCES

Samsung Semiconductor Newsroom, "Eight Major Semiconductor Processes, Part 1: The Wafer" (in Korean) · Samsung Semiconductor, "Eight Major Semiconductor Processes, Part 2: Oxidation, Which Protects the Wafer Surface" (in Korean) · Samsung Semiconductor Newsroom, "Eight Major Semiconductor Processes, Part 6: Deposition and Ion Implantation" (in Korean) · ETRI, *Electronics and Telecommunications Trends* vol. 38 no. 6, "Complementary FET—The Future of the Semiconductor Transistor" (in Korean) · ScienceON JAKO202509150403114, "EUV lithography Technology Process Trends and Research Prospects" (in Korean) · ScienceON JAKO201326940561717, "Atomic Layer Deposition: Overview and Applications" (in Korean) · ASML, "5 things you should know about High NA EUV lithography" (2024) · van Schoot et al. (ASML, ZEISS), "High-NA EUV lithography enabling Moore's law in the next decade," EUVL Workshop 2017 · TSMC press release (2019-10-07), N7+ EUV in volume production · Deal & Grove (1965), doi:10.1063/1.1713945

Learning objectives

1. Recast the eight major front-end processes not as a list of separate items but as one repeating loop.
2. Explain the three terms of the Rayleigh criterion and describe how the industry attacked each one.
3. Explain why the moves to FinFET and GAA were physically unavoidable.
4. Explain the physical difficulties of EUV (reflective optics, the light source, stochastic defects).
5. Tell apart which physical quantity each process controls.

0. Self-study guide — for readers working alone

> **THIS IS THE LONGEST AND MOST PHYSICAL SESSION**
>
> If the four sessions before it were about "how much," this one is about "how." Do not be put off by the length. **The only genuinely hard part is section 4 (lithography); the rest either prepares you for section 4 or follows from it.** If time is short, reading sections 3 and 4 alone gets you 80 percent of this session.

What you bring from earlier sessions

FROM EARLIER SESSIONS	IN THIS SESSION
Session 4 $Y = e^{-AD_0}$	The roughly 1,000 process steps that produce that D_0
Session 4 reticle limit	Where that limit comes from: the exposure field
Session 3 "Why is HBM expensive?"	Here, why logic chips are expensive

Prerequisite check — what you can skip and what you must know

CONCEPT	ONE-LINE ANSWER	NEED
MOSFET	A switch: apply a voltage to the gate and a current path (the channel) opens between source and drain	★ Required
Doping	Si has 4 outer electrons and pairs up neatly with its 4 neighbors. Mix in phosphorus (P) or arsenic (As), which have 5, and one electron is left over: n-type. Mix in boron (B), which has 3, and a vacancy (a hole) appears: p-type. Picture a restaurant whose tables for 4 are all full. A party of 5 arrives and one person ends up standing and wandering about; a party of 3 arrives and the guests shuffle along one seat at a time to fill the empty place. Pure Si is not an insulator either; it is a semiconductor with low conductivity	☆ Concept only (section 7)
Wavelength and frequency	The shorter light's wavelength, the more energy each photon carries. $E = hc / \lambda$	★ Required (Exercise 3)
Numerical aperture NA	How wide an angle of light a lens gathers. The larger it is, the better the resolution	★ Required
Diffraction limit	The narrowest line light can draw is proportional to the wavelength and inversely proportional to NA ($k_1\lambda$ / NA). Techniques that shrink the process factor k_1 let you draw lines narrower than the wavelength: in Exercise 1, about 43 nm with 193 nm light	★ Required
Plasma	A gas separated into electrons and ions. In etching it produces reactive ions	☆ Concept only
Annealing and diffusion	Using heat to repair the crystal damaged by ion implantation and to seat the dopants on lattice sites (activation). Apply heat for long and the dopants spread out (diffusion)	☆ Concept only (section 7)

If you have a background in signal processing or communications, there is a shortcut. Read "Seen as signal processing" in section 4 first. Lithography is ultimately **the problem of**

passing a signal through a low-pass filter and then recovering the original, and OPC (optical proximity correction) is **pre-distortion.** Once you have that mapping, all of section 4 reads in a familiar language.

Reading order (160 minutes recommended; it is fine to split it over two sittings)

STEP	SECTION	TIME	NOTE
Day 1			
1	Sections 1 and 2, intuition and the wafer	20 min	
2	**Section 3, how the transistor structure evolved**	30 min	The conceptual highlight
3	**Section 4, lithography**	**45 min**	**The heart of this session.** Do Exercises 1, 2 and 3 here
Day 2			
4	Sections 5–7, etching, deposition, ion implantation	30 min	Much easier than section 4
5	Section 8, inspection and metrology	15 min	The most enjoyable section if your background is in CS or AI
6	Section 9, putting it together, plus Exercises 4 and 5	20 min	

Never skip

1. **The Rayleigh criterion in section 4** — Exercises 1, 2 and 5 all come from it.
2. **"The brutal physics of EUV" in section 4** — the background to Exercise 3, and the real reason EUV is so expensive.
3. **Exercise 4, step yield** — it takes five minutes and sums up the difficulty of semiconductor manufacturing in one number.

A common sticking point — "Why does a larger NA make depth of focus worse with the square?"

Put the two equations side by side and the tension shows.

$$\mathrm{CD} = k_1 \frac{\lambda}{\mathrm{NA}}, \qquad \mathrm{DOF} = k_2 \frac{\lambda}{\mathrm{NA}^2}$$

Intuitively it goes like this. A large NA means **the light is gathered from a very wide angle.** Light converging from a wide angle comes to a sharper point at focus (resolution ↑), but move a little away from focus and the rays arriving from different directions fall badly out of step (depth of focus ↓). At a distance z from focus, the path difference between an oblique ray at the edge and a ray arriving straight down is roughly $z(1-\cos\theta) \approx z\sin^2\theta \,/\, 2$. In air $\sin\theta = \mathrm{NA}$, so the path difference is proportional to the square of NA; once it exceeds a set fraction of the wavelength the image blurs, so the allowable z is proportional to $\lambda \,/\, \mathrm{NA}^2$. In the same medium, doubling NA halves the CD and cuts the depth of focus to 1/4.

As a photography analogy: open the aperture wide (NA ↑) and the image is sharp, but **the depth that stays in focus becomes paper-thin.** A DOF of a few tens of nanometers for High-NA EUV means that the unevenness of the wafer surface has to fit inside it, and that is why **CMP (planarization) becomes a required process, not an option.**

---

1. Getting the intuition — chipmaking is "printing and coating," not "sculpting"

The whole picture

If you think of making a chip as sculpting (carving away), you will get it wrong. It is closer to **multilayer printing.**

1. Coat the silicon disc (the wafer) with a thin film (**deposition**)
2. Apply a light-sensitive material and draw a pattern with light (**lithography**)
3. Develop it to dissolve away the areas that were (or were not) exposed
4. Cut away only the exposed parts (**etching**)
5. Remove the remaining light-sensitive material
6. Smooth the surface flat (**CMP**)
7. Inspect (**metrology and inspection**)
8. → Back to step 1 for the next layer

The loop is **repeated dozens of times,** and each pass contains many detailed steps, so the whole process exceeds **1,000 steps.** From the time a wafer goes into the fab until the finished product comes out takes **several months.**

Why there are so many layers

A chip has two main parts.

- **FEOL (front-end of line):** the part that builds transistors on the silicon surface. The device layer.
- **BEOL (back-end of line):** the part that stacks many layers of wiring on top to connect the transistors. The latest chips have 15–20 wiring layers.

> In a city analogy, FEOL is the buildings and BEOL the road network. And **lately the road network is the bigger problem**: as wires grow thinner their resistance rises steeply, until it dominates signal delay and power consumption. This is the background to techniques such as backside power delivery (BSPDN).

---

2. The wafer — the starting point

How it is made

1. **Refining polycrystalline silicon** — metallurgical-grade silicon is obtained from quartzite and refined to ultra-high purity by the Siemens process or similar. A purity of **around 11N (99.999999999%)** is required, because a single impurity changes device characteristics.
2. **Growing a single crystal (the Czochralski method)** — dip a seed crystal into molten silicon and draw it up slowly while rotating it; the atoms take on the seed's crystal orientation, and a huge single-crystal ingot grows.
3. **Slicing → lapping → etching → polishing** — the ingot is cut into thin slices and the surface polished flat to the atomic level.
4. **Epitaxy (optional)** — a high-quality silicon layer that continues the crystal orientation is grown on top of the wafer.

Why it stopped at 300 mm

The larger the wafer, the more dies you get at once and the lower the cost per die. So wafers grew from 150 to 200 to 300 mm. But **the move to 450 mm ran aground.**

The reasons were purely economic. Making the wafer diameter 1.5 times larger multiplies the area by 2.25, but every tool that handles the larger wafer has to be built anew (lithography tools, etchers, transport systems, all of them), and obtaining uniform process results across the whole wafer becomes much harder. **The development cost exceeded the savings.** Instead of larger wafers, the industry chose scaling and the third dimension (stacking).

> **General lesson:** technology choices in the semiconductor industry are decided not by physical possibility but by **cost-effectiveness.** The failure of 450 mm is the classic example.

Names on the wafer

Look down on a wafer that has finished processing and you see the same square repeated like a checkerboard. Each part has a name.

- **Chip (die):** one square of the checkerboard. It is the result of the lithography tool printing the same pattern square after square.
- **Scribe line:** the narrow lanes between dies. When processing is done, the wafer is cut along these lines (back end, Session 6).
- **TEG (test element group):** test devices and patterns placed in spare spots such as the scribe lines. They are used to measure whether line width, resistance and threshold voltage came out on target, and they do not go into the product.
- **Edge dies:** lay squares on a round disc and incomplete squares appear around the rim. The second term of the dies-per-wafer formula in Session 4 is exactly this loss. The edge

exclusion width written in wafer specifications (example from a 300 mm specification: 3 mm) is also an unusable band.

- **Orientation mark:** a mark that tells the tools the crystal orientation and the angle at which the wafer sits. Suppliers' specifications show that wafers of 150 mm or less usually have a **flat**, an edge cut off in a straight line, while 200 and 300 mm wafers have a **notch**, a V-shaped groove. The notch in a 300 mm specification is 1.0 mm deep, with a 90° angle, in the <110> direction (the same specification gives a diameter of 300±0.1 mm and a thickness of 775±25 μm).

Thermal oxidation — the first film silicon grows from itself

It is missing from the loop in section 1, but many processes start here: oxidizing the silicon surface to grow a SiO_2 film. If deposition (section 6) piles a different material on top, thermal oxidation **turns the silicon itself into an oxide film.** As the film grows, the silicon underneath is consumed.

- **Conditions:** usually done at high temperatures of 800–1,200 °C. Using oxygen (O_2) alone is **dry** oxidation; adding water vapor (H_2O) is **wet.** Dry oxidation grows slowly, but the film is dense with good electrical properties, so it is used for thin films. Wet oxidation grows several times thicker at the same temperature and time, at the price of lower density.
- **Uses:** insulating films between devices, films that mask unwanted areas during ion implantation, films that protect the layer beneath during etching.

Thickness and time — the Deal–Grove equation. The oxidant diffuses through the oxide already grown and reacts at the interface with the silicon. Whichever of the two processes is slower sets the rate. Deal and Grove (1965) combined the two in one equation.

$$x^2 + Ax = B(t + \tau)$$

x is the oxide thickness, t the oxidation time, and τ a correction that converts any film present from the start into time. A and B are coefficients set by the temperature and the oxidant (O_2 or H_2O). The original paper showed that oxidation data for both oxygen and water vapor at 700–1,300 °C fit this equation well.

The two limits of the equation are the two phases of growth.

- **When thin** — ignore x^2 and $x \approx (B / A)(t + \tau)$. Thickness is proportional to time (the linear regime). The interface reaction sets the rate.
- **When thick** — grow long enough that x^2 dominates, and $x^2 \approx Bt$, so $x \approx \sqrt{Bt}$. This is the parabolic regime, where diffusion through the film sets the rate. Doubling the thickness takes four times as long.

The actual values of A and B are set separately for each temperature and oxidant and used from tables. This book does not give them.

---

3. How the transistor structure evolved — from planar to FinFET and GAA

What a MOSFET does (a 30-second review)

A channel lies between source and drain; apply a voltage to the gate and charge is induced in the channel so current flows. The gate is a **valve.** A good transistor is one whose valve shuts tight (small off-state leakage) and opens fully (large on-state current).

The problem — the short-channel effect

When scaling shortens the channel, source and drain come closer together. Then **the drain voltage begins to act directly on the channel.** The drain, not the gate, ends up controlling the channel, and current leaks even with the gate turned off (DIBL, a lowering of the threshold voltage).

In the valve analogy: the pipe has become so short that even with the valve shut, pressure from the other side pushes straight through.

The remedy — wrap more of the channel with the gate

The gate's control is proportional to **the area over which the gate faces the channel.** So the structure evolved like this.

STRUCTURE	FACES THE GATE WRAPS	PERIOD	FEATURES
Planar	1 face (top)	–2011	Simple, vulnerable to the short-channel effect
FinFET	**3 faces** (top + both sides)	2011–	The channel stands up like a fin and the gate wraps around it. Drive strength set by the number of fins (quantized)
GAA / nanosheet	**4 faces (all around)**	2022–	The channel is made as horizontal sheets fully surrounded by the gate. **Sheet width can be varied continuously for fine-tuning of drive strength**
(Next generation) CFET	4 faces + n/p stacked vertically	Research stage	nFET and pFET stacked one above the other to save area

Two practical advantages of GAA

1. **Better electrostatic control** — wrapping all 4 faces lets the gate win even in a short channel.
2. **Design flexibility** — in FinFETs drive strength could be set only by the number of fins (an integer), so there was a quantization loss. GAA lets the **width** of the nanosheets vary **continuously,** so circuit designers get exactly the drive strength they want.

The price: to build a stack of several nanosheets, you grow a Si/SiGe superlattice, selectively remove only the SiGe so the sheets float free, and fill the gaps with the gate dielectric and metal. **Atomic layer deposition (ALD), which coats narrow gaps evenly, is the key to this process** (see section 6).

Backside power delivery (BSPDN) — the next branch point

Traditionally, power and signals shared the wiring layers on the front of the wafer. But power wires need to be thick (to prevent IR drop) while signal wires need to be dense, so they fight over space.

The remedy: move the power wiring to the **back** of the wafer. Flip the wafer, grind it thin, build a power network on the back, and connect it directly to the transistors with nano-TSVs.

The gain: relief of front-side wiring congestion (better area and performance) + more efficient power delivery (less IR drop). **The price:** wafer thinning, alignment, changed heat-dissipation paths and more; the difficulty rises sharply.

As of October 2026, Intel was first to apply it, under the name PowerVia, in 18A. The first 18A product (Core Ultra series 3) was announced in January 2026. TSMC is introducing it in A16 as Super Power Rail. When it announced A16 in April 2024, it planned production for 2026. At the April 2026 technology symposium a TSMC executive was reported as saying: "A16 will be ready for production in 2026. However, actual product ramp depends on customers, and we expect volume production to begin in 2027." This schedule does not appear in TSMC's press releases or earnings materials, and as of October 22, 2026 there is no official announcement that volume production has begun. **This is the point where the data center power story of Session 7 comes down to the level of the transistor.**

---

4. Lithography — the part we go deepest into

The Rayleigh criterion

The smallest resolvable line width (CD, critical dimension) is

$$\mathrm{CD} = k_1 \frac{\lambda}{\mathrm{NA}}$$

- λ : wavelength of the light source
- NA : numerical aperture = $n\sin\theta$. How wide an angle of light the lens gathers
- k_1 : process factor. Its theoretical lower bound for a single exposure is about 0.25

And the depth of focus (DOF) is

$$\mathrm{DOF} = k_2 \frac{\lambda}{\mathrm{NA}^2}$$

The central tension: a larger NA improves resolution linearly but **makes depth of focus worse with the square.** If the surface is even slightly uneven, the pattern fails to form. This is why CMP (planarization) is a required process (section 7).

Seen as signal processing — a shortcut for readers with a statistics or computing background

Treat the lithography system as **a low-pass filter in a linear shift-invariant system** and everything falls into place.

- The mask pattern is a signal in the spatial domain. Its Fourier transform gives its spatial-frequency components.
- The lens (projection optics) is **a low-pass filter with cutoff frequency** $f_c \propto \mathrm{NA} / \lambda$**.** Spatial frequencies above that cannot pass through the lens pupil and are lost.
- The image formed on the wafer is therefore **the low-pass-filtered mask pattern.** Sharp corners (high-frequency components) are smeared.

Seen in this frame:

- **OPC (optical proximity correction):** distorting the input (the mask) in advance to get the output you want. **A textbook inverse problem, or deconvolution.** Serifs are added to corners and line widths are fine-tuned.
- **ILT (inverse lithography technology):** formulate the whole thing as an optimization problem and solve for the mask pattern. The objective function is "the difference between the pattern on the wafer and the target pattern"; the variables are the mask pixels. **It is a large non-convex optimization problem, and today it is solved on GPU clusters.**
- **Phase-shift mask (PSM):** reverses the phase of adjacent openings by 180° to create destructive interference at the boundary and raise contrast. In other words, it controls phase as well as amplitude.
- **Source-mask optimization (SMO):** optimizes the shape of the illumination (the illumination distribution in the pupil) together with the mask.

> **Going further:** "computational lithography" is, as the name says, a large computing problem. **It is a leading example of machine learning put to work in real industry,** and one of the most natural entry points into the semiconductor industry for people trained in CS or AI.

A history of shortening the wavelength

LIGHT SOURCE	WAVELENGTH	PERIOD
g-line (mercury lamp)	436 nm	–early 1990s
i-line	365 nm	1990s
KrF excimer	248 nm	late 1990s
ArF excimer	193 nm	2000s
ArF immersion	193 nm (about 134 nm in water)	2007–
EUV	**13.5 nm**	2019–

The immersion trick: in $\mathrm{NA} = n\sin\theta$, n is the refractive index of the medium. Fill the space between lens and wafer with water (n≈1.44) instead of air (n≈1) and NA can be pushed past 1,

up to 1.35. **It was a clever way to enlarge NA without changing the light source, and it let 193 nm hold out for another ten years or so.** The 134 nm in the table is the wavelength shortened in water, $\lambda_0 / n = 193 / 1.44$. That effect is already contained in the n of NA $= n\sin\theta$, so the Rayleigh equation takes the vacuum wavelength of 193 nm and an NA of 1.35 (Exercise 1). Plug in 134 nm and you count the refractive index twice.

Reduction projection — the mask is larger than the wafer

A lithography tool does not print the pattern of the mask (reticle) at the same size; it prints it **reduced.** A 0.33 NA EUV tool (ASML NXE) reduces it 4 times in both directions, and 4 times is also the standard for today's ArF scanners. That means the pattern on the mask is drawn 4 times larger than the pattern to be drawn on the wafer.

Two things follow from this setup. First, pattern errors on the mask are also carried to the wafer at roughly 1/4 the size (in very fine patterns the mask error can be carried over amplified beyond that; this is called the mask error enhancement factor, MEEF). Dust on the patterned surface of the mask likewise prints small. Second, a defect on the mask is nonetheless **repeated identically in every field printed with that mask.** Unlike particles that fall at random on the wafer (the Poisson defects of Session 4), a single mask defect creates a defect at the same spot in every field on the wafer.

The field printed in one exposure is at most 26×33 mm on the wafer. Do the arithmetic: at 4× reduction the pattern area on the mask is 104×132 mm, and that fits inside a mask 6 inches (152.4 mm) on a side.

High-NA has different magnifications in the two directions. Raising NA to 0.55 makes the mirrors larger, which makes the angle at which light strikes the mask larger, and then the reflective mask can no longer reflect the pattern properly. Reducing 8 times in both directions solves the angle problem but forces a larger mask. ASML's answer is **anamorphic optics, which reduce 4 times in one direction and 8 times in the other.** Do the arithmetic: the 104×132 mm area on the mask stays the same, and only the field in the 8× direction becomes 132 ÷ 8 = 16.5 mm, so the field on the wafer is halved to 26×16.5 mm. **The field is halved so that the same 6-inch mask can still be used.**

Multi-patterning — a way around k_1

Once you reach the lower bound of k_1 for a single exposure (about 0.25), you print one layer in several exposures.

- **LELE (litho-etch-litho-etch):** split the pattern between two masks and expose and etch twice
- **SADP / SAQP (self-aligned double / quadruple patterning):** deposit spacers on the sidewalls of an existing pattern and remove the original pattern, and **a pattern at half the pitch** remains. Repeat to go down to 1/4

The price is high: the number of process steps doubles, cost and cycle time jump, and **overlay error** becomes a new yield variable. **The real motive for adopting EUV was not "smaller" but "fewer multi-patterning steps, and so lower cost."**

The brutal physics of EUV

A wavelength of 13.5nm goes beyond ultraviolet and is close to soft X-rays. Here everything gets hard.

① **No lenses** 13.5nm is absorbed by almost every material. Refractive lenses are impossible, so **only reflective optics** will do. Even the mirrors are not ordinary mirrors but **multilayer Bragg reflectors,** alternating molybdenum and silicon layers a few nanometers thick, stacked up to about 100 layers, and even so they reflect at most about 70% of the light. Pass through several mirrors and the losses multiply, so **only a tiny fraction of the source's output reaches the wafer.**

② **A vacuum is required** Air absorbs EUV too, so the whole light path must be in vacuum.

③ **The light source is extremely hard** Droplets of tin (Sn) are dropped tens of thousands of times a second, and each is hit twice by a high-power CO_2 laser (flattened by a pre-pulse, then hit by the main pulse) to make a plasma that emits 13.5nm light. **The conversion efficiency from electricity to EUV is very low, so the tool consumes a lot of power.** On top of that, spattered tin contaminates the collector mirror. The one number that became the threshold for volume production is source power. When TSMC announced its first EUV volume-production process in 2019, it said its EUV tools had "output power of greater than 250 watts for day-to-day operations" (the throughput calculation in Exercise 3 starts from this power).

④ **The mask is reflective too** Transmissive masks cannot be used, so reflective masks are used; because light strikes them at an angle, a new distortion called **shadowing** appears. And since a protective membrane over the mask (pellicle) absorbs EUV, developing pellicles was a long-standing problem.

⑤ **Stochastic defects — statisticians, take note**

This is EUV's most interesting problem.

- Photon energy $E = hc / \lambda$ is inversely proportional to wavelength, so a 13.5nm photon carries about **14 times the energy** of a 193nm photon.
- To deliver the same exposure energy (dose), **you need correspondingly fewer photons.**
- Photon arrival is a Poisson process, so the relative fluctuation of the count N is $\sigma / \mu = 1 / \sqrt{N}$. When N is small, the fluctuation is large.
- The result: some spots happen to receive fewer photons and develop too little; others receive more. **Line edge/width roughness (LER/LWR), missing holes and bridge defects** occur at random.

These are not defects that better processing can remove; they are a statistical inevitability. The countermeasure is to raise the dose (more photons), but then throughput falls. **The three-way trade-off between resolution, throughput and stochastic defects** is the fundamental constraint of the EUV process. New resists (metal-containing resists and others) are attempts to improve this balance.

> **Deeper, for statisticians:** the problem is the tail of the defect probability. A chip has billions of features, so even if the failure probability of an individual feature is on the order of 10^{-12}, it affects yield at the chip level. **This is an industrial problem where extreme value theory is actually used.**

High-NA EUV — still unfolding

- The next generation of EUV, with NA raised from 0.33 to **0.55.** ASML TWINSCAN **EXE:5200.**
- As of October 2026: Intel was the first to install the second-generation EXE:5200B and passed acceptance testing, and in July 2026 ASML and Intel announced that Intel was producing and shipping in high volume some Core Ultra series 3 products with certain 18A layers printed by High-NA. Yields are said to be on par with those printed on the existing NXE. SK hynix brought a production EXE:5200B into its M16 fab in Icheon, Korea, in September 2025, and imec received an EXE:5200 in March 2026 and plans to complete qualification by the fourth quarter of 2026.
- **For now it is used on some layers only.** Intel has said it will validate it on 18A and then extend it to 14A production. Which node other companies will start using High-NA at, and how many tools have been installed worldwide, could not be confirmed from official announcements as of October 2026.

Two prices of High-NA

1. **A steep drop in depth of focus** — since DOF $\propto \lambda$ / NA^2, raising NA from 0.33 to 0.55 cuts DOF to about $(0.33/0.55)^2 \approx 0.36$ times. Flatness requirements become extreme.
2. **Anamorphic optics and a smaller field** — High-NA uses anamorphic optics with different magnifications in the two directions (4× and 8×), and as a result **the field that can be exposed at once is halved (26×16.5mm).** The existing reticle limit of 858 mm² becomes 429 mm². Why it halves was worked out in "Reduction projection" above.

> ★ **This is the link between Session 5 and Session 6:** when the exposure field shrinks, a chip larger than the field cannot be printed in one exposure. There are two ways out: stitching, which prints two fields edge to edge, and chiplets, which build the chip in pieces and join them in the package (Exercise 5). Stitching pays for alignment error at the seam and for design constraints, so the more layers are printed with High-NA, the stronger the incentive to split large chips into chiplets. It continues in Session 6.

---

5. Etching — the subtractive process

Think of copperplate etching. Coat a copper plate with a film that acid will not dissolve, scratch a picture through it with a needle, and dip it in the etching bath: only the scratched, exposed places dissolve, leaving grooves. In semiconductors, the photoresist is that film, lithography and development are the needle, and etching is the bath.

Wet vs. dry

- **Wet etching:** immersion in a chemical solution. It is isotropic (it eats in equally in all directions), so it is unsuited to fine patterns. Used for cleaning, removing sacrificial layers and the like.
- **Dry etching (plasma):** reactive ions accelerated by an electric field strike vertically. It can achieve **anisotropy,** so it is the standard for fine patterns.

Two key measures

1. **Anisotropy:** you must etch only straight down. Etch sideways (undercut) and the line width collapses. Plasma etching gets its verticality by combining ion bombardment (physical) with chemical reaction, and forms a protective film on the sidewalls (passivation) to stop sideways etching.
2. **Selectivity:** the ratio of etch rates between the material to be removed and the material to be left. With low selectivity you punch through into the layer below. In the GAA process, **removing only the SiGe while leaving the Si nanosheets** is the textbook case demanding extreme selectivity.

Other factors managed are **etch rate and uniformity.** Etch rate is the depth removed per unit time, and it sets how many wafers one tool processes per hour. Uniformity is whether the center and the edge of the wafer are etched to the same depth. If uniformity is poor, then while the under-etched places are being finished off, the places that broke through first keep eating into the layer below, and correspondingly more has to rest on selectivity.

High-aspect-ratio (HAR) etching — the hardest of the hard

This is the job of drilling deep, narrow holes.

- **3D NAND:** channel holes running through hundreds of layers. The aspect ratio exceeds 100:1.
- **TSVs:** vias running through the silicon wafer (Session 6).
- **DRAM capacitors:** keeping the capacitance while shrinking the area means digging deep.

Why it is hard: the deeper you go, the harder it is for reaction products to get out; ions bounce off the sidewalls and lose their direction; and you get bowing, where the hole widens near the top first, and taper, where it narrows toward the bottom. So special techniques such as **cryogenic etching and pulsed plasma** are brought in.

---

6. Deposition — the additive process

METHOD	PRINCIPLE	FEATURES	USES
PVD (physical vapor deposition)	Target material is sputtered or evaporated and flies onto the surface	Fast, poor step coverage	Metal wiring, barriers
CVD (chemical vapor deposition)	Gaseous precursors react to form a film on the surface	Good coverage, may need high temperatures	Insulating films, polysilicon
ALD (atomic layer deposition)	**Self-limiting reactions** repeated in alternation	**Thickness control at the atomic-layer level, excellent step coverage**	Gate dielectrics, barriers, nanosheet gates

Why ALD is the savior of scaling

ALD introduces precursor A and lets it react only until the surface is saturated (self-limiting), purges, then introduces precursor B to complete the reaction. Because the reaction stops once the surface is saturated, the thickness deposited per cycle (growth per cycle) is nearly constant. That value is set by the size of the precursor molecules, the surface condition and the temperature, and it is usually less than one atomic layer.

Two properties that result

1. **Thickness is controlled precisely by the number of cycles.** With a constant growth per cycle, the repetition count can set thickness down to the ångström.
2. **Even complex shapes are coated evenly.** Allow each cycle enough time for the precursor to reach inside holes and between gaps and for the leftover gas to be cleared out (purged), and the inside of a deep hole and the narrow gaps between nanosheets are coated to nearly the same thickness. The more complex the shape, the longer each cycle takes.

GAA relies on ALD. The gate dielectric and metal have to fill the gaps of a few nanometers between floating nanosheets uniformly, and ALD, which builds up one layer at a time with surface reactions that stop by themselves, is the best fit for coating such narrow gaps evenly.

High-k gate dielectrics — a case of quantum mechanics changing an industry

The thinner the gate dielectric (traditionally SiO_2), the better the gate's control, but **below 1–2nm electrons pass through the dielectric by quantum tunneling and leakage current explodes.**

The remedy: in the capacitance $C = \varepsilon A / d$, instead of reducing d, use a material with a large ε (permittivity). With high permittivity you get the same capacitance from a physically thicker film, blocking tunneling while keeping control. Hafnium oxide (HfO_2)-based materials play this role, and polysilicon gates were also replaced by metal gates (**HKMG: high-k metal gate**, introduced around 2007).

> **Going further:** this is a case where a textbook result of quantum mechanics, "the tunneling probability depends exponentially on the barrier thickness," forced an industrial transition worth billions of dollars.

---

7. Ion implantation, annealing and CMP

Ion implantation

Doping in practice is not elegant. **Dopant ions (boron, phosphorus, arsenic and others) are accelerated to tens to hundreds of keV and physically driven into the silicon.**

- **Depth control:** set by the acceleration energy. The higher the energy, the deeper they go.
- **Concentration control:** set by the ion beam current and time (dose).
- **Distribution:** stopping depth is random, so the concentration profile is roughly Gaussian (strictly, a skewed distribution). **It is predicted by Monte Carlo simulation** — a tool familiar to statisticians, used as is.
- **Side effect:** the ions break up the crystal lattice. So **annealing** is needed to restore the crystal and to seat the dopants on lattice sites, activating them electrically.
- **The tension:** anneal for long and the crystal recovers well, but the dopants diffuse and the profile smears. In fine devices this is fatal, so annealing has moved to ever shorter and more intense forms such as **rapid thermal annealing (RTA), millisecond annealing and laser annealing.**

CMP (chemical mechanical polishing)

Why it is required: as section 4 showed, depth of focus is shallow. If the surface is uneven, the pattern will not form in the next layer's exposure. Unevenness accumulates as layers are stacked, so **every layer** has to be planarized.

How: a pad rubs the surface mechanically while a slurry that softens it chemically flows over it. It is a combination of chemistry and mechanics, hence the name CMP.

What is hard: materials polish at different rates, producing **dishing** (wide metal areas becoming concave) and **erosion.** So **dummy fill** patterns that even out pattern density are inserted at the design stage. **This is a textbook case of design adapting to the manufacturing process (DFM).**

Cleaning

It is easy to mention in passing, but in reality a substantial share of all process steps is cleaning. One particle is one defect, and one defect is yield lost (the D_0 in $Y = e^{-AD_0}$!), so cleaning goes in between every process.

---

8. Inspection and metrology — the direct point of contact for CS and AI people

Inspection vs. metrology

- **Inspection:** is there a defect? Where? (particles, bridges, missing features)
- **Metrology:** are the dimensions on target? (line width, film thickness, overlay error)

Main techniques

TECHNIQUE	PRINCIPLE	FEATURES
Optical inspection	Scattered light, contrast	Fast; limited in detecting defects below the wavelength
E-beam inspection	Scanning electron beam	High resolution, **very slow**
OCD / scatterometry	Fits the diffracted-light spectrum to a model to back out the shape	Nondestructive, **solves an inverse problem**
Overlay metrology	Measures alignment marks	Manages layer-to-layer alignment error
CD-SEM	Measures line width directly with an electron microscope	Accurate, slow, local

Why this is AI's main stage

1. **The data is overwhelming.** Thousands of dies on one wafer, tens of thousands of wafers a day in a fab. Wafer maps, sensor logs and metrology values pile up in vast quantities.
2. **Defect classification is a classic classification problem.** Telling defect types apart in SEM images is image classification itself.
3. **Wafer-map patterns point to causes.** Spatial patterns such as rings, scratches and edge concentration correspond to particular tool or process problems. **A spatial-statistics plus pattern-recognition problem.**
4. **Virtual metrology.** Not every wafer can be measured (slow and expensive). Tool sensor data are used to predict metrology values and cut down sampling. **It is a regression problem, and uncertainty quantification really matters.**
5. **Predictive maintenance.** Predicting tool failures in advance.
6. **APC (advanced process control).** A control loop that feeds metrology results back into the process conditions for the next wafer. **Run-to-run control is a classic time-series control problem.**

> **Going further:** the place where AI is used most, and most for real, in the semiconductor industry is not inside the chip but in **running the fab.**

"If you can't measure it, you can't make it"

Metrology itself is becoming a bottleneck of scaling. Measuring structures a few nanometers across nondestructively, quickly and accurately keeps getting harder. 3D structures (GAA nanosheets, 3D NAND channels) are harder still because they cannot be seen from the surface.

9. Putting it all together — how one layer is made

Take one wiring layer (BEOL) as an example (the damascene process):

```
1. Deposit insulating film (CVD)
2. Coat photoresist → expose → develop      ← defines the pattern
3. Etch trenches/vias (plasma)
4. Strip photoresist + clean
5. Deposit barrier/seed layer (ALD/PVD)     ← prevents Cu diffusion
6. Fill with copper by electroplating
7. Remove excess copper + planarize by CMP  ← needed before the next layer can be exposed
8. Deposit cap layer
9. Inspect and measure
   ↓ repeat for the next layer (15–20 times)
```

The wiring material changed once in a big way. Until the 1990s, wiring was aluminum (Al). An aluminum film was laid down and etched so that only the wiring shapes remained, and the narrow contact holes connecting transistors to the wiring were filled with tungsten (W) (W plugs). Aluminum in direct contact with silicon intermixes with it at the boundary and ruins the junction, so a barrier metal was laid between them. As wires grew thinner, aluminum hit two walls. One was resistance; the other was electromigration (current gradually pushing metal atoms along until the wire breaks or bunches up). Copper has lower resistance and resists this effect, but its atoms leak into silicon and ruin the devices, and it is hard to etch cleanly. So copper came in not by etching it into shape but by the damascene method, which **cuts grooves into the insulating film, fills them, and removes the excess with CMP.** Filling the via and the wiring groove in one go is dual damascene, and the barrier in step 5 above stops copper from leaking out. IBM announced on September 22, 1997 that it would put copper-wired chips into volume production (first shipments in September 1998), and said that copper wires conduct electricity with about 40% less resistance than aluminum. In the 2000s copper became the industry standard for wiring.

The sense to take away

- Even one layer has this many steps, and they repeat dozens of times.
- If any single step fails, that wafer is (partly) lost. **Even if each of 1,000 steps succeeds 99.99% of the time, the whole comes to** $0.9999^{1000} \approx 90\%$**.** That tells you how extreme the yield required of each step is.
- With cycle times of several months, feedback is slow even when a problem is found. This is **why virtual metrology and in-line control matter.**

Mask count and cost

The finer the node, the more masks (number of layers × the multi-patterning factor), the more process steps and the more tool time, and the wafer price rises. **EUV is a tool for easing this increase by reducing multi-patterning.** That is why "EUV is expensive" and "EUV cuts costs" can both be true.

10. Common misconceptions

Misconception 1: "Lithography is just printing, like taking a photograph" → It makes structures smaller than the wavelength, so physically it is a fight against the diffraction limit. The pattern on the mask looks completely different from the pattern on the wafer (OPC/ILT).

Misconception 2: "With EUV, multi-patterning is no longer needed" → EUV has its own single-exposure limit, so at the leading edge EUV multi-patterning is done. High-NA is an attempt to postpone it.

Misconception 3: "Scaling will soon end because of the laws of physics" → The **economic limit** arrives before the physical one: the point where you can make it but cannot sell it at that cost. The failure of the 450mm wafer is an example.

Misconception 4: "GAA is simply better than FinFET" → Process complexity and cost rise sharply. There is no reason for every product to move to GAA, and mature nodes will remain.

Misconception 5: "Defects can be eliminated by raising cleanliness" → True for particle defects, but **defects that are a statistical inevitability,** such as EUV's stochastic defects, are not solved by cleanliness.

Misconception 6: "The bigger the wafer, the better" → As the failure of 450mm shows, the cost of redeveloping the tools and the difficulty of achieving uniformity can exceed the savings.

Misconception 7: "Ion implantation is doping by chemistry" → The ions are physically driven in. That is why the lattice is damaged and annealing must always follow.

Misconception 8: "EUV is used only at the leading edge, 3 nm and below" → EUV volume production began at the 7 nm class. TSMC billed N7+ as "the industry's first commercially available Extreme Ultraviolet (EUV) lithography technology" and began volume production in the second quarter of 2019 (Samsung Electronics announced in October 2018 that it had begun wafer production on its 7LPP process using EUV), and in October of that year it announced that customer products were reaching the market in high volume. In the same announcement it said that its successor N6 would apply EUV more widely, for 18% higher logic density than N7, and ASML explains that its 0.33 NA tools are used to print the critical layers of the 7, 5 and 3 nm nodes. As section 4 showed, the motive for bringing in EUV lay less in "smaller" than in cutting multi-patterning steps.

11. Quantitative exercises

Exercise 1 — The Rayleigh criterion With $k_1 = 0.3$, find the CD for (a) ArF immersion (λ=193nm, NA=1.35), (b) EUV (λ=13.5nm, NA=0.33) and (c) High-NA EUV (λ=13.5nm, NA=0.55). What is the improvement factor from one generation to the next?

Exercise 2 — The price in depth of focus Under the same conditions, find the DOF ratios (assume the same k_2). How much stricter do flatness requirements become with High-NA?

Exercise 3 — Stochastic defects At the same exposure energy, find the ratio of photon counts at 193nm and 13.5nm ($E = hc / \lambda$), and calculate how many times larger the relative fluctuation $1 / \sqrt{N}$ becomes. By what factor must the dose rise to reach the same level of fluctuation? What happens to throughput then?

Exercise 4 — Step yield If the whole process has 800 steps, what must the average yield of each step be to reach a target line yield of 90%? And with 1,200 steps?

Exercise 5 — Exposure field and die size Find the largest die area that can be made with the existing reticle limit (about 26×33mm) and with the High-NA field (about 26×16.5mm), and compare the yields of the largest dies by applying the yield equation of Session 4. **It becomes the starting point of the chiplet discussion in Session 6.**

---

12. Discussion questions

1. Of the three terms of the Rayleigh criterion, which has the most room for improvement from here? Are Hyper-NA or a new wavelength realistic?
2. How does the field reduction of High-NA EUV change design methodology? Do chiplets become a requirement rather than an option?
3. For defects that are statistically inevitable, such as stochastic defects, what strategy should the industry take? Which is more efficient: leaving a margin at the design level, or raising the dose in the process?
4. If metrology is becoming a bottleneck of scaling, how far can methods of "control without measurement" (model-based control, virtual metrology) go? How is their reliability assured?
5. If scaling runs into an economic limit, what is the next axis of performance improvement? (A preview of Session 6)

---

13. Further study

- **Standard textbooks:** Plummer, Deal & Griffin, *Silicon VLSI Technology* — the standard text on process physics. May & Spanos, *Fundamentals of Semiconductor Manufacturing and Process Control* (Wiley, 2006, doi:10.1002/0471790281) — **it covers statistical process control, so it suits readers with a statistics background especially well**
- **The original paper on thermal oxidation:** Deal & Grove, "General Relationship for the Thermal Oxidation of Silicon," *J. Appl. Phys.* 36(12):3770–3778 (1965), doi:10.1063/1.1713945 — shows what experimental data the equation in section 2 came from
- **Device physics:** Sze & Ng, *Physics of Semiconductor Devices* (Wiley, 3rd ed. 2006, doi:10.1002/0470068329, a full-scale textbook), Taur & Ning, *Fundamentals of Modern VLSI Devices* (Cambridge, 2nd ed. 2009, doi:10.1017/CBO9781139195065, short-channel effects and scaling theory)
- **Lithography:** Chris Mack, *Fundamental Principles of Optical Lithography* (Wiley, 2007, doi:10.1002/9780470723876) — **written in terms of Fourier optics, so it reads very well if you have a signal-processing background.** The author's website lithoguru.com has online lectures and basic explanations
- **EUV:** Bakshi (ed.), *EUV Lithography* (SPIE, 2nd ed. 2018, doi:10.1117/3.2305675). For the anamorphic design of High-NA, van Schoot et al., EUVL Workshop 2017 presentation and ASML, "5 things you should know about High NA EUV lithography"
- **Latest developments:** presentations at IEDM, the VLSI Symposium and SPIE Advanced Lithography; imec's published roadmap material
- **AI × semiconductor manufacturing:** run a model yourself on WM-811K, the public dataset for wafer-map defect classification (Wu, Jang, Chen, *IEEE Trans. Semiconductor Manufacturing* 28(1) (2015), doi:10.1109/TSM.2014.2364237), and section 8 will come alive

Self-study appendix A — solutions to the quantitative exercises

Solution to Exercise 1 — The Rayleigh criterion

▾ Solution

CD = $k_1\lambda$ / NA, with $k_1 = 0.3$ held fixed.

GENERATION	λ	NA	CD
(a) ArF immersion	193 nm	1.35	$0.3 \times 193 / 1.35 =$ **42.9 nm**
(b) EUV	13.5 nm	0.33	$0.3 \times 13.5 / 0.33 =$ **12.3 nm**
(c) High-NA EUV	13.5 nm	0.55	$0.3 \times 13.5 / 0.55 =$ **7.4 nm**

Improvement factors

STEP	FACTOR	WHAT CONTRIBUTED
(a) → (b)	**3.49×**	Almost entirely the wavelength (193 → 13.5, 14.3×). NA actually **got worse,** 1.35 → 0.33
(b) → (c)	**1.67×**	Entirely NA (0.33 → 0.55)

How to read it — this table tells you everything about the character of EUV

In moving to EUV, NA was **given up.** 13.5 nm is absorbed by every material, so refractive lenses cannot be used and only reflective optics are possible, and it is hard to build a large NA from mirrors. The wavelength gained 14.3 times and NA lost 4.1 times, for a net gain of 3.5 times.

And **High-NA is the work of winning back exactly that lost NA.** 0.33 → 0.55 is a mere 1.67 times, yet for it the whole optical system is redesigned and the field is halved (Exercise 5).

> **Caution:** $k_1 = 0.3$ is near the practical lower bound for a single exposure (theoretical lower bound 0.25). The 42.9 nm is actually a half-pitch, and finer patterns made with ArF immersion are printed in several exposures by **multi-patterning** — that is, k_1 is bypassed, not lowered.

What about real tools? You can work k_1 backward from the resolution the toolmaker states. ASML gives the resolution of its High-NA tool (EXE) as 8 nm. The arithmetic gives $k_1 = 8 \times 0.55 / 13.5 = 0.326$. With the 13 nm resolution of the 0.33 NA tool (NXE), $k_1 = 13 \times 0.33 / 13.5 = 0.318$. In other words, the $k_1 = 0.3$ assumed in the table above is not far from real tool specifications.

Solution to Exercise 2 — The price in depth of focus

▾ Solution

DOF $= k_2\lambda$ / NA^2. Keep k_2 the same and compare λ / NA^2 alone.

GENERATION	λ / NA^2	RELATIVE DOF	WITH $k_2 = 0.5$
(a) ArF immersion	193 / 1.8225 = 105.9	1.00	53 nm
(b) EUV	13.5 / 0.1089 = 124.0	**1.17**	62 nm
(c) High-NA EUV	13.5 / 0.3025 = 44.6	**0.42**	**22 nm**

One unexpected result: EUV (b) actually has **17% more** depth of focus than ArF immersion (a), thanks to its lower NA. EUV's arrival did not suddenly raise flatness requirements.

The real problem is High-NA. DOF shrinks to 0.36 times (b) and 0.42 times (a), to **about 22 nm.**

What has to fit inside 22 nm

- Local unevenness of the wafer surface (CMP's job)
- Flatness and vibration of the wafer chuck
- The thickness of the resist film itself — a 50 nm resist already exceeds the DOF

What follows in a chain

1. **CMP requirements become extremely strict.** Planarization becomes a precondition of lithography (section 7).
2. **The resist must be thinner.** But a thin resist has **less etch resistance** — it disappears before it has done its job as a mask.
3. So **hardmasks and multilayer resists** are used, or new materials that hold up even when thin, such as **metal-oxide resists.**

What to take away: a single improvement in resolution pushes up requirements in a chain, through planarization, materials and etching. **Why scaling is not "just a matter of changing the lithography tool"** is all in this one exercise.

The figure below shows the case of EUV (13.5 nm, NA 0.33) with $k_1 = 0.3$ and $k_2 = 0.5$. In the screen edition you can change the light source and NA of the three generations, k_1 and k_2, and you can also work k_1 backward from a tool's nominal resolution.



Figure 5-1. At $k_1 = 0.3$, CD is 42.9 nm for ArF immersion, 12.3 nm for EUV and 7.4 nm for High-NA EUV. Depth of focus with $k_2 = 0.5$ is 53, 62 and 22 nm, so High-NA gets 0.36 times the EUV value. — In the HTML edition you can change the values here (HTML edition, Session 5, Solution to Exercise 2).

Solution to Exercise 3 — Stochastic defects

▾ Solution

The energy of one photon is $E = hc / \lambda$, so it is inversely proportional to wavelength.

$$\frac{E_{\text{EUV}}}{E_{\text{ArF}}} = \frac{193}{13.5} = 14.3$$

An EUV photon carries **14.3 times the energy** of an ArF photon (about 92 eV vs. 6.4 eV).

Photon count at the same dose (energy per unit area)

With the same energy and each photon 14.3 times larger, the **number** of photons is only 1/14.3 = **7%.**

$$N_{\text{EUV}} = \frac{N_{\text{ArF}}}{14.3}$$

Relative fluctuation

In Poisson statistics the relative fluctuation of a count N is $1 / \sqrt{N}$, so

$$\frac{\sigma_{\text{EUV}} / N_{\text{EUV}}}{\sigma_{\text{ArF}} / N_{\text{ArF}}} = \sqrt{14.3} = 3.78 \text{ times}$$

To reach the same level of fluctuation, the dose must rise 14.3 times. (N has to become 14.3 times larger.)

Throughput at that point

EUV source power directly sets the processing time per wafer. Other conditions being equal, such as the time to move and align the wafer, the time to expose one wafer is proportional to the dose divided by the source power.

$$t_{\text{exposure}} \propto \frac{\text{dose}}{\text{source power}}, \qquad \text{throughput} = \frac{1}{T_0 + t_{\text{exposure}}}$$

Here T_0 is the fixed time that does not depend on dose, such as wafer transport and alignment. So throughput is not simply proportional to source power ÷ dose; as the exposure time gets shorter, it saturates against 1 / T_0.

The reference point is the day-to-day source power of more than 250 W seen in section 4 (TSMC's 2019 announcement). Keep that power and raise the dose 14.3 times, and the exposure time becomes 14.3 times longer. In the limit where exposure time dominates everything, **wafers per hour collapse to 1/14.3.** With a fixed time T_0 the drop is smaller, and how much smaller depends on how large T_0 is compared with the original exposure time t. Throughput becomes $(T_0 + t)$ / $(T_0 + 14.3t)$ of the original, so it falls to 13% if $T_0 = t$, to 45% if $T_0 = 10t$, and hardly at all if T_0 is much longer than the exposure time. This exercise gives no T_0, so read 1/14.3 as the worst case. In that limit the productivity of a tool costing hundreds of millions of dollars drops to 7%, which is economically impossible. Conversely, keeping throughput would require raising source power 14.3 times, from 250 W to about 3.6 kW by the arithmetic.

So what is actually done

MEASURE	CONTENT
Raise the dose **only partly**	ASML rates EUV tool throughput at 30 mJ/cm² (NXE:3600D, 3800E) and 50 mJ/cm² (High-NA EXE:5000). The dose is not raised 14.3-fold to match the photon count; it stays in the tens of mJ/cm²
Improve resist sensitivity and chemistry	So that one photon triggers more reactions (chemical amplification, metal-oxide resists)
Avoid stochastic defects **by design**	Designs built mainly on regular one-dimensional patterns, avoidance of forbidden pitches
Catch them with metrology	Stochastic defects are rare and scattered, so fast large-area inspection is needed → section 8

This is what "the brutal physics of EUV" amounts to. Shortening the wavelength bought resolution, but in exchange **the photon count itself became scarce and statistical uncertainty grew.** There is a trade-off in principle between resolution and stochastic defects, and it ties dose, throughput and yield together.

Connection: these stochastic defects add straight onto the D_0 of Session 4. And as D_0 rises, the die area allowed for the same target yield shrinks ($A_{50} = \ln 2 \,/\, D_0$). **The physics of lithography sets the size of the chip.**

The figure below shows the case where the EUV dose is raised by 193/13.5, that is about 14.3 times, to match the fluctuation of ArF. In the screen edition you can vary the dose multiple and the reference source power and see the relative fluctuation, the exposure time, and the source power needed to keep the same exposure time. Because the fixed time T_0 is unknown, throughput is shown only as its limiting value, where exposure time dominates everything.

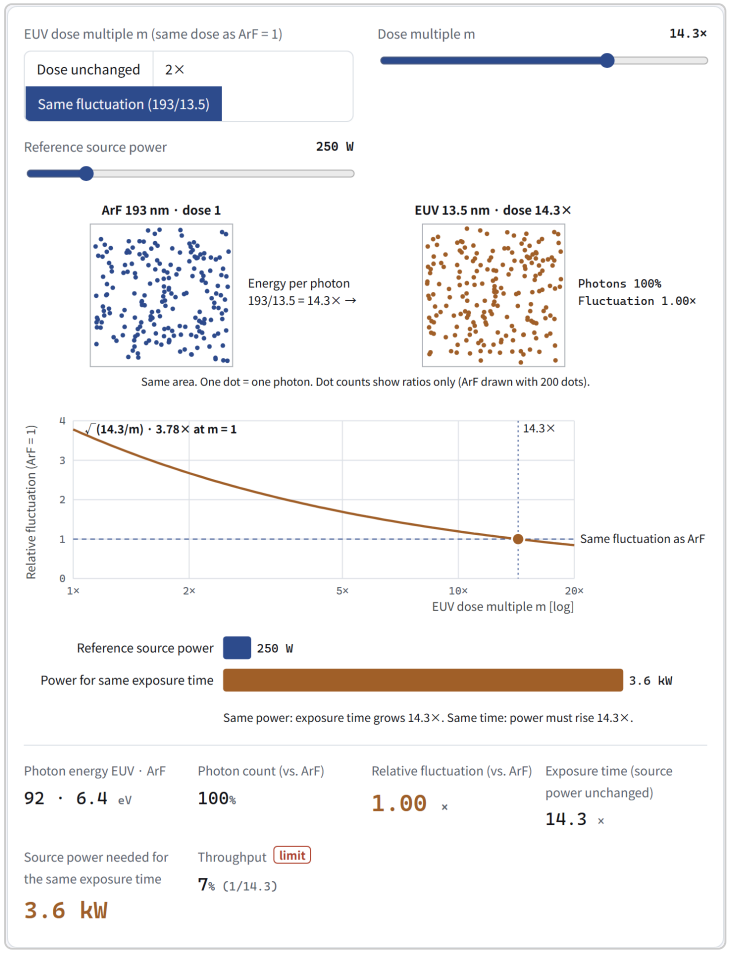


Figure 5-2. At the same dose, EUV delivers 7% as many photons as ArF, and the relative fluctuation is 3.78 times larger. Raising the dose 14.3 times to match the fluctuation makes exposure 14.3 times longer at the same source power; holding exposure time instead requires source power to rise from 250 W to about 3.6 kW. — In the HTML edition you can change the values here (HTML edition, Session 5, Solution to Exercise 3).

Solution to Exercise 4 — Step yield

▾ Solution

If the total yield is the product of the step yields, $y^n = Y$, so $y = Y^{1/n}$.

800 steps, target 90%

$$y = 0.9^{1/800} = \exp\left(\frac{\ln 0.9}{800}\right) = \exp\left(\frac{-0.10536}{800}\right) = 0.9998683$$

$1 - 0.9998683 = 1.317 \times 10^{-4}$, so **the allowable defect rate per step is 131.7 ppm.** If a single step slips just 132 times in a million, the target is missed.

1,200 steps, same target

$$y = \exp\left(\frac{-0.10536}{1200}\right) = 0.9999122 \;\Rightarrow\; 87.8 \text{ ppm per step}$$

When the steps increase 1.5 times, the allowable defect rate shrinks to 2/3.

Seen the other way round, it is far more frightening

YIELD PER STEP	DEFECT RATE PER STEP	LINE YIELD, 800 STEPS	1,200 STEPS
99.99%	100 ppm	$0.9999^{800} = 92.3\%$	88.7%
99.9%	1,000 ppm	$0.999^{800} = 44.9\%$	30.1%
99%	10,000 ppm	$0.99^{800} = 0.03\%$	Almost 0

The line yields in the table are y^n calculated directly. When the defect rate $p = 1 - y$ is small, $y^n = (1-p)^n \approx e^{-pn}$, so at 800 steps you can also approximate 99.99% as $e^{-0.08}$, 99.9% as $e^{-0.8}$ and 99% as e^{-8}. At the number of digits shown in the table, the two calculations agree.

99% per step is by common sense an excellent figure, yet after 800 steps virtually nothing comes out.

What this exercise sums up

Here is why semiconductor manufacturing differs fundamentally from other manufacturing. **Hundreds of steps multiply in series, so each step has to be perfect not "almost" but "to the ppm."** So

- six-sigma-level control is demanded of every single process
- inspection and metrology go in at every step (section 8)
- a decision to add steps (multi-patterning!) becomes not a simple cost increase but **an increase in yield risk**

The real cost of multi-patterning: section 4 said multi-patterning bypasses k_1. This is its price — print one layer in 4 exposures and the steps increase, and by the table above the line yield is cut multiplicatively. **When EUV is adopted despite its expense, the reason is often not resolution but the reduction in step count.**

The figure below shows the case of a target line yield of 90% at 800 steps. In the screen edition you can change the number of steps and the target yield, and you can also go the other way and find the line yield from a yield per step.

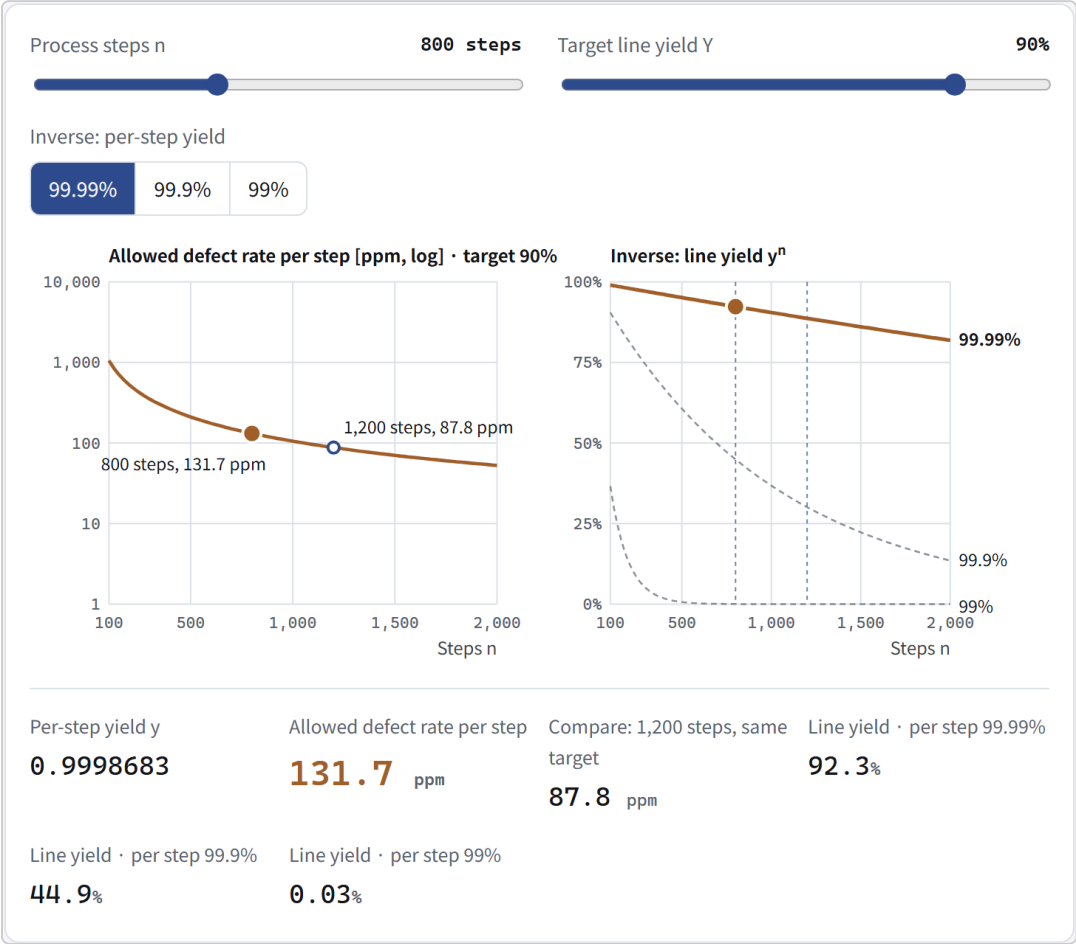


Figure 5-3. To reach 90% line yield over 800 steps, the allowed defect rate per step is 131.7 ppm; over 1,200 steps, 87.8 ppm. — In the HTML edition you can change the values here (HTML edition, Session 5, Solution to Exercise 4).

Solution to Exercise 5 — Exposure field and die size

▾ Solution

EXPOSURE METHOD	FIELD SIZE	LARGEST DIE AREA
Existing (including 0.33 NA)	26 × 33 mm	**858 mm²**
High-NA EUV	26 × 16.5 mm	**429 mm²**

High-NA uses anamorphic optics (the reduction ratio differs in one direction only), so

the field is halved.

Why exactly half? The calculation in "Reduction projection" in section 4 is the answer. Existing tools reduce 4 times in both directions, so 104×132 mm on the mask becomes 26×33 mm on the wafer. High-NA reduces 8 times in one direction while using the same 6-inch mask, so the field in that direction shrinks to 132 ÷ 8 = 16.5 mm. Only the direction whose magnification doubled is halved, exactly.

Applying the yield equation of Session 4 ($D_0 = 0.1$)

FIELD	AREA	$\lambda = AD_0$	YIELD
Largest die, existing	8.58 cm²	0.858	**42.4%**
Largest die, High-NA	4.29 cm²	0.429	**65.1%**

The dilemma shows clearly

High-NA **lets you draw finer patterns but halves the area you can draw at once.** Yield improves (42% → 65%), but the chips you can make get smaller.

Yet AI accelerators keep getting bigger. The direction is dies of around 800 mm² with 8 HBM stacks attached, and such a die cannot be printed in one exposure with the High-NA field.

There are two remedies

REMEDY	CONTENT	PROBLEM
Stitching	Join two fields to make one die	Alignment error and defects at the seam, more steps
Chiplets	Make small dies from the start and join them in the package	Interface overhead, assembly yield

Conclusion: a die larger than the High-NA field must either pay the cost of stitching or be split into chiplets. Which is cheaper is decided by weighing the alignment and yield cost of the seam against the interface and assembly cost of chiplets, but **the more layers are printed with High-NA, the stronger the incentive to lean toward chiplets.** It amounts to adding the High-NA field constraint as one more cost item to the chiplet profit-and-loss calculated in Exercise 4 of Session 4. Session 6 continues that accounting.

The figure below shows what field the same pattern area on a 6-inch mask (104 × 132 mm) becomes on a 0.33 NA tool and on a High-NA tool, and the yield of the largest die when $D_0 = 0.1$. In the screen edition you can change the optics and D_0.

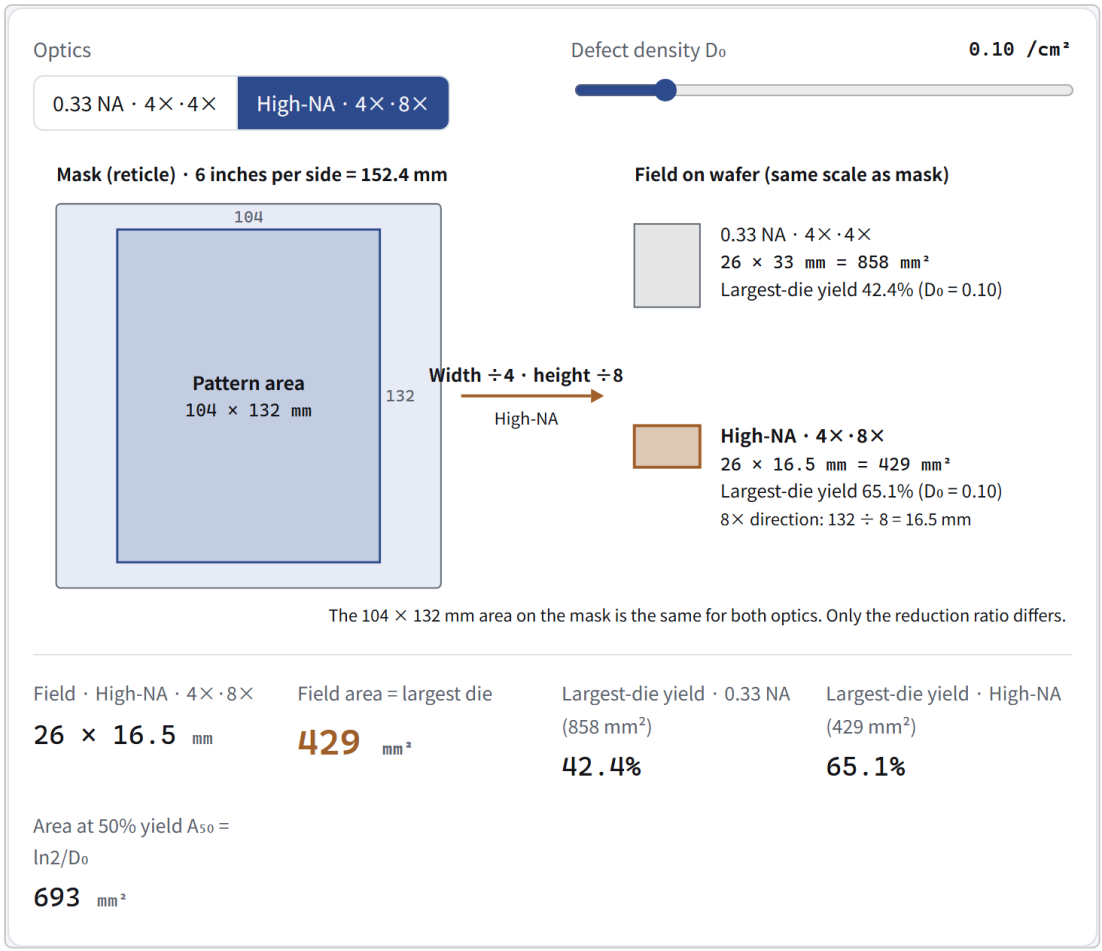


Figure 5-4. Shrinking the 104 × 132 mm pattern area on the mask 4× in each direction gives a 26 × 33 mm wafer field (858 mm²); High-NA, which shrinks one direction 8×, gives 26 × 16.5 mm (429 mm²). With D_0 = 0.1, the maximum die yields are 42.4% and 65.1%. — In the HTML edition you can change the values here (HTML edition, Session 5, Solution to Exercise 5).

Self-study appendix B — leads for the discussion questions

The numbers are those of the discussion questions in section 12. Rows marked + add questions worth thinking about alongside them.

QUESTION	CORE AXIS	LEADS
1. Which of the three terms has room left	**Room left vs. price**	k_1 is close to the theoretical single-exposure lower bound of 0.25 (0.3 assumed in Exercise 1; 0.32–0.33 worked back from tool specifications). Push NA beyond 0.55 and depth of focus shrinks in inverse proportion to NA² (Exercise 2), while the field and mask problems grow again. Shorten the wavelength further and each photon carries more energy, so the photon count at the same dose falls further (Exercise 3). For any of the terms, the starting point is to calculate the price that comes with it before the gain that is left

QUESTION	CORE AXIS	LEADS
2. Field reduction and design methodology	**Stitching vs. chiplets**	A die larger than the High-NA field (429 mm^2) must be printed in joined fields by stitching or split into chiplets (Exercise 5). "Is it a requirement?" turns into a comparison of the alignment and yield cost of the seam against the interface and assembly cost of chiplets. Print even one layer with High-NA and the field constraint applies at that layer, so the number of layers printed with High-NA changes the answer
3. Stochastic defects — dose vs. design margin	**Throughput vs. area**	Raising the dose shrinks the relative fluctuation in proportion to $1 / \sqrt{\text{dose}}$, but exposure time grows in proportion to the dose (Exercise 3). A design margin enlarges features or uses regular patterns to thin the tail of the failure probability, at the cost of area. Convert both into the same unit, "the cost of a good die from one wafer," and you can compare them with the cost-per-die formula of Session 4
4. Control without measurement	**Prediction vs. verification**	Virtual metrology predicts metrology values from tool sensors and shrinks the measurement sample (section 8). The way to find out whether a prediction was wrong is, in the end, measurement too, so rather than eliminating measurement it is used to concentrate measurement on the wafers whose predictions are most uncertain. Reliability comes from continually checking the distribution of prediction errors against real measurements
5. The next axis after scaling	**Which limit comes first**	There are three candidates: (i) physics (channels counted in atoms, stochastic defects), (ii) economics (gain relative to mask and tool cost), (iii) heat (power density). The ones reached first today are (ii) and (iii); (i) still has room. **That is why the industry is shifting its axis from scaling to integration (packaging)**
+ How far can you go without EUV	**Step count vs. resolution**	Multi-patterning can keep up on resolution itself. What breaks is **economics** — as Exercise 4 showed, more steps cut yield multiplicatively, and more masks and longer cycle times drive costs up steeply. The accurate statement is not "technically impossible" but "economically impossible"
+ What backside power delivery means	**Separating wiring from devices**	Dedicating the front to signals and the back to power is not a simple improvement but **a change in the philosophy of layer placement.** Exercise 4 (IR drop) in Session 7 checks quantitatively why it matters as much as GAA
+ The role of AI in inspection and metrology	**Detection vs. root-cause analysis**	Classification of defect images is already mature. The hard problem is **tracing back from wafer-map patterns to the tool and step that caused them,** and it ties directly to the clustering (α) of Session 4. Seen as a spatial-statistics plus causal-inference problem, it opens up
+ Where should localization start	**The hierarchy of substitutability**	As the three-layer structure of Session 1 showed, the less replaceable a layer, the harder it is to produce domestically. But difficulty and payoff are not proportional — **materials and parts that are substitutable but whose supply is concentrated** offer the most for the cost

Self-study appendix C — key terms

TERM	ONE-LINE DEFINITION
Front-end process (FEOL + BEOL)	The whole of the wafer processing that builds transistors (FEOL) and the wiring layers above them (BEOL) on the wafer. Distinguished from the back end, which cuts and joins wafers
Thermal oxidation	A process that grows a SiO_2 film by oxidizing silicon at high temperature. The thickness follows the Deal–Grove equation

TERM	ONE-LINE DEFINITION
Reduction projection	Printing the mask pattern onto the wafer at reduced size. 4× is standard; High-NA is 4× and 8×
Dual damascene	A wiring process that cuts vias and wiring grooves into the insulating film, fills them with copper in one go, and removes the excess by CMP
CD (critical dimension)	The smallest line width that has to be made
Rayleigh criterion	$\mathrm{CD} = k_1\lambda$ / NA. The fundamental limit on resolution
Numerical aperture (NA)	A measure of the angle over which a lens gathers light
Depth of focus (DOF)	The range of depth over which focus holds. $\propto \lambda$ / NA^2
EUV (extreme ultraviolet)	Lithography with extreme ultraviolet light of 13.5 nm wavelength
High-NA EUV	The next generation of EUV, with NA raised from 0.33 to 0.55. The field is halved
Multi-patterning	Printing one layer in several exposures to get around the resolution limit
Stochastic defect	A random defect arising from a shortage of photons
OPC (optical proximity correction)	Deforming the mask so as to cancel diffraction distortion in advance
Photoresist	A light-sensitive film whose properties change when exposed to light
Etching	The process of removing unwanted material
Anisotropy / selectivity	The degree to which etching cuts only vertically / the degree to which only the intended material is cut
HAR etch (high aspect ratio etch)	Etching that drills deep, narrow holes. A hard problem in 3D NAND and TSVs
Deposition	The process of building up the needed films
ALD (atomic layer deposition)	Deposition that builds up atomic layer by atomic layer for precise thickness control
High-k	A gate dielectric with high permittivity. It stops leakage current
Ion implantation	A process that drives impurity ions in to change electrical properties
CMP (chemical mechanical polishing)	A process that planarizes the surface chemically and mechanically
GAA (gate-all-around)	A transistor structure in which the gate surrounds the channel on all sides
CFET (complementary FET)	A next-generation structure that stacks n-type and p-type vertically
BSPDN (backside power delivery network)	Moving the power wiring to the back of the wafer
Inspection / metrology	Looking for whether there are defects / measuring dimensions and thicknesses

Self-study appendix D — self-assessment

☐ I can write the two Rayleigh equations and explain that **resolution and depth of focus trade off** → section 4 ★

☐ I can show by calculation that EUV gained 14 times on wavelength and lost 4 times on NA → Exercise 1

☐ I can say why the DOF of High-NA is around 22 nm and what that demands of CMP and resists → Exercise 2 ★

☐ I can explain **why stochastic defects become a problem in EUV** using photon counts and $1 / \sqrt{N}$ → Exercise 3 ★

☐ I can say in numbers why 99% per step is a disaster → Exercise 4 ★

☐ I can explain that the real cost of multi-patterning is **yield,** not resolution → Exercise 4

☐ I can explain why the field reduction of High-NA strengthens the incentive for chiplets, comparing it with stitching → Exercise 5 ★

☐ I can string together the motives for planar → FinFET → GAA → CFET in one line → section 3

☐ I can state the difference between inspection and metrology and explain where AI is mainly used → section 8

Before the next session

This session was **the story of making one chip.** Session 6 is **the story of joining many chips.**

QUESTION THIS SESSION LEFT	THE ANSWER IN SESSION 6
How do you make a chip larger than the High-NA field?	Print it in joined fields by stitching, or split it into chiplets and join them in the package
What connects the split chips?	Interposers, TSVs, hybrid bonding
Splitting means more wiring; what about power?	Exercise 2 — wire length sets the energy
Where do the HBM stacks go?	Exercise 3 — the interposer area exceeds the reticle limit

---

Sources

Introductory material chosen by the study curriculum (the "front-end processes" item)

CURRICULUM SUBTOPIC	LINK	HOW THIS BOOK USES IT
Wafer	Samsung Semiconductor Newsroom, "Eight Major Semiconductor Processes, Part 1: What Is a Wafer, the Core Material of Integrated Circuits?" (in Korean)	Basics of wafer manufacturing in section 2. **The eight-processes series is the most accessible introductory material in Korean**
Transistor	ETRI, *Electronics and Telecommunications Trends* vol. 38 no. 6, "Complementary FET—The Future of the Semiconductor Transistor" (in Korean)	A review of CFET (vertically stacked n/p) technology trends. Background to the last row (CFET) of the structure-evolution table in section 3
Transistor	ScienceON patent KOR1020250044769, "Multi-layered Vertical Gate-All-Around (GAA) Transistor and Method of Fabricating the Same" (Korean patent document)	Patent literature on stacked GAA structures. Reference for GAA and CFET in section 3
Lithography	Hana Financial Investment, semiconductor industry report "Semiconductors: Knowledge Is Power — EUV" (2019-04-23, in Korean, PDF)	An industry report on EUV lithography tools. **An investment report, but its explanation of EUV technology is detailed and its figures are good, so it is useful as presentation material for section 4.** It dates from 2019, though, so the High-NA part needs newer material
Lithography	Korea Security Agency of Trade and Industry, journal no. 8, Jeon Youngmin, "Advanced Semiconductor EUV Lithography Technology and International Security Situation" (2024, in Korean, PDF)	A paper that treats the technical peculiarities of EUV lithography together with the export-control situation (Korea–Japan, U.S.–China). Links EUV in section 4 with the supply-chain discussion of Session 1
Lithography	ScienceON JAKO202509150403114, "EUV lithography Technology Process Trends and Research Prospects" (in Korean)	Academic literature on EUV process technology trends. Section 4
Etching	ScienceON JAKO198965349552573, "An Analysis of Trends in Dry Etching Technology" (in Korean)	Academic literature on dry etching technology (from 1989 — for understanding the principles; current processes need other material)
Etching	ScienceON patent KOR1019990043077, "APPARATUS FOR ETCHING SEMICONDUCTOR" (Korean patent document)	A patent on etching equipment
Deposition	Samsung Semiconductor Newsroom, "Eight Major Semiconductor Processes, Part 6: How Do You Give a Semiconductor the Electrical Properties It Needs?" (in Korean)	Basics of deposition and ion implantation in sections 6 and 7
Deposition	ScienceON JAKO201326940561717, "Atomic Layer Deposition: Overview and Applications" (in Korean)	Academic literature giving an overview of ALD. ALD in section 6
CMP	ScienceON report KAR2006021584, "Mechanism and Friction Characteristics of Chemical Mechanical Polishing (CMP)" (in Korean)	A report on the mechanism of CMP. Background to section 7
Inspection and metrology	ScienceON DIKO0015747253, "A Study on Quality Control in Semiconductor Metrology and Improving the Detection Power of Particle Inspection Equipment" (in Korean)	A thesis on metrology quality control and particle inspection. Section 8

CURRICULUM SUBTOPIC	LINK	HOW THIS BOOK USES IT
Ion implantation	Samsung Semiconductor official site, "Eight Major Semiconductor Processes, Part 6: Giving Semiconductors Their Electrical Properties — Deposition and Ion Implantation" (in Korean)	The official-site version (Korean) of the same series as Part 6 in the newsroom above. Part 2 of the same series covers oxidation

Primary sources added in this revision (checked 2026-10-08)

- Samsung Semiconductor, "Eight Major Semiconductor Processes, Part 2: Oxidation, Which Protects the Wafer Surface" (in Korean) — thermal oxidation in section 2 (800–1,200 °C, dry and wet)
- Deal, B. E. & Grove, A. S., "General Relationship for the Thermal Oxidation of Silicon," *J. Appl. Phys.* 36(12):3770–3778 (1965), doi:10.1063/1.1713945 — the Deal–Grove equation in section 2 (bibliographic data from Crossref; the equation and temperature range checked in the abstract. The coefficient tables were not checked)
- Silicon Valley Microelectronics, 300 mm wafer specifications SV025 · SV027 — notch, diameter, thickness and edge exclusion in section 2. The same supplier's specifications for 150 mm and smaller (SV001–SV013) have a flat; those for 200 mm (SV017–SV020) have a notch
- van Schoot et al. (ASML, ZEISS), "High-NA EUV lithography enabling Moore's law in the next decade," EUVL Workshop, 2017-06-15 — reduction ratios in section 4 (0.33 NA 4×/4×, 0.55 NA 4×/8×), fields of 26×33 mm and 26×16.5 mm, 6-inch mask
- ASML, "5 things you should know about High NA EUV lithography" (2024) — reason for the anamorphic optics, the halved field, EXE resolution of 8 nm, in section 4
- ASML, EUV lithography systems page — NXE resolution of 13 nm, use at the 7, 5 and 3 nm nodes
- ZEISS SMT, "EUV lithography: The light behind the digital age" — mirrors in section 4
- ASML TWINSCAN NXE:3600D · ASML TWINSCAN NXE:3800E · ASML TWINSCAN EXE:5000 — doses at which throughput is rated, solution to Exercise 3
- ASML press release (2026-07-15), "High NA EUV reaches new readiness milestone with first high-volume Logic product" · Intel second-quarter 2026 earnings press release (2026-07-23) — status of High-NA in volume production, section 4
- imec press release (2026-03-18), "Imec receives the world's most advanced High NA EUV system" · SK hynix Newsroom (2025-09-03), "SK hynix Introduces Industry's First Commercial High NA EUV" · Intel Newsroom (2024-04-18), "With High NA EUV, Intel Foundry Opens New Frontier in Chipmaking" — status of High-NA, section 4
- TSMC press release (2019-10-07), "TSMC's N7+ Technology is First EUV Process Delivering Customer Products to Market in High Volume" — source power above 250 W in section 4, misconception 8 in section 10
- IBM History, "Copper interconnects" — copper wiring in section 9 (announced 1997-09-22, about 40% lower resistance than aluminum, dual damascene)
- TSMC press release (2024-04-24), A16 announcement · Tom's Hardware (2026-04-22),

technology symposium report: A16 volume production in 2027 (quoting a TSMC executive) · TSMC 2nm (N2) technology page · Intel Foundry VLSI Symposium press release (2026-06-16) · Intel fourth-quarter and full-year 2025 earnings press release (2026-01-22) — status of backside power delivery, section 3

Verification status

- Status of High-NA EUV (section 4) — rechecked 2026-10-08. Intel passing the first acceptance test of an EXE:5200B and shipping in high volume, from July 2026, products with certain 18A layers printed by High-NA (ASML press release 2026-07-15, Intel second-quarter earnings press release); SK hynix bringing an EXE:5200B into M16 (newsroom 2025-09-03); imec receiving an EXE:5200 with qualification planned by Q4 2026 (press release 2026-03-18); Intel's plan to apply it in 14A production (newsroom 2024-04-18). Statements such as "a dozen or so installed worldwide," "volume-production use expected 2027–2028" and "TSMC A14 and Samsung SF1.4 are the target nodes" were not confirmed from primary sources and are not included
- Reduction ratios and exposure fields (26×33 mm, High-NA 26×16.5 mm, 4×, 4×8×) — confirmed in the ASML/ZEISS 2017 presentation and ASML's 2024 article (2026-10-08). The 104×132 mm area on the mask is calculated from these values
- EXE resolution 8 nm, NXE 13 nm — confirmed on ASML's official pages (2026-10-08). The k_1 worked backward in Exercise 1 (0.326, 0.318) is calculated from these values
- TSMC N7+ (which TSMC billed as "the industry's first commercially available" EUV technology, volume production from Q2 2019), N6 applying EUV more widely for 18% higher logic density, source power above 250 W in day-to-day operation — checked against the TSMC press release original (2026-10-08; wording rechecked 2026-10-22). Samsung Electronics' start of 7LPP EUV wafer production confirmed in a Samsung Electronics press release (2018-10-18)
- IBM's copper wiring announced 1997-09-22, about 40% lower resistance than aluminum — confirmed on IBM's official history page (2026-10-08; the 40% figure rechecked 2026-10-22). The explanations of W plugs, barrier metal and electromigration are standard textbook content
- Thermal oxidation temperature (800–1,200 °C) and the difference between dry and wet — confirmed on Samsung Semiconductor's official site. The Deal–Grove equation and its range of application (700–1,300 °C, O_2 and H_2O) — confirmed in the original paper's abstract (2026-10-08). The two limits follow directly from the equation
- The 300 mm wafer notch (1.0 mm deep, 90°, <110>), diameter, thickness and 3 mm edge exclusion — confirmed in the supplier's specifications (SVMI SV025, SV027) (2026-10-08). **The original SEMI M1 standard (paid) was not checked (SEMI M1 original not compared)**
- Backside power delivery: Intel 18A applying PowerVia, first product announced January 2026; TSMC A16 Super Power Rail (production planned for 2026 when announced in 2024 → volume production expected to begin in 2027, according to a reported TSMC executive's remarks at the April 2026 technology symposium) — rechecked 2026-10-08 against each company's press releases; the A16 schedule rechecked 2026-10-22 (Tom's Hardware 2026-04-22; TSMC's third-quarter earnings press release of 2026-10-15 does not mention A16).

Same sources as the table in section 4 of Session 4

- EUV wavelength 13.5nm, reflective multilayer mirrors, tin-droplet plasma source — standard technical facts
- Make-up of the EUV mirrors (alternating molybdenum and silicon layers, up to about 100 layers, each a few nanometers thick) and their reflectivity (up to 70%) — confirmed on ZEISS SMT's technology page ("By stacking up to 100 layers, each just a few nanometers thick"; "mirrors that reflect up to 70 percent of the light") (2026-10-22)
- Doses at which EUV throughput is rated (NXE:3600D and 3800E 30 mJ/cm^2, EXE:5000 50 mJ/cm^2) — confirmed on ASML's product pages (2026-10-22). The dose used in an actual process varies with the layer and the resist
- The Rayleigh criterion and the depth-of-focus equation — standard optics
- Evolution of the transistor structure (planar → FinFET → GAA) — confirmed 2026-08-02
- Doping (pentavalent and trivalent impurities, the semiconducting nature of intrinsic silicon) — textbook standard (Sze & Ng)

Unconfirmed and reserved (not asserted in the text)

- **When TSMC A16 enters volume production** — the 2027 schedule is confirmed only by reports of a TSMC executive's remarks in April 2026 and does not appear in TSMC's press releases or earnings materials. There is still no official announcement that volume production has begun (checked 2026-10-22)
- **Number of High-NA tools installed worldwide, nodes at which companies other than Intel apply it** — not confirmed from official announcements as of October 2026
- **Values of the Deal–Grove coefficients A and B** — not given in the text because the tables of the original paper were not checked
- **Overall efficiency of EUV optics** — varies with the number of mirrors and the generation. The text asserts only the qualitative point that "the losses multiply"
- **Wafer purity of 11N** — widely quoted, but it varies with use and grade
- **Number of process steps (over 1,000) and cycle time (several months)** — vary widely with product and node. Given only as orders of magnitude
- **Theoretical lower bound of k_1, 0.25** — a value from single-exposure imaging theory; the actual process factor varies with illumination and mask techniques
- k_1, k_2, step counts and the like in the exercises of section 11 — **assumed values for calculation practice**

SESSION 6

Back-End Processing and Advanced Packaging

KEY TAKEAWAY

If you want to understand AI semiconductors, this is the session that matters most. The back end, once a low-value finishing step of "cut, attach and wrap," has become the main stage for performance gains. The reason is that three pressures acted at once: ① **yield** — a large die makes yield collapse as $Y = e^{-AD_0}$; ② **the exposure field limit** — there is a ceiling on the area that can be printed in one shot, and High-NA halves it; ③ **heterogeneous optimization** — logic, SRAM, analog and I/O each have a different best process. All three point to "split it up and join the pieces." And the HBM problem of Session 3 (to pull out a very wide bus you have to sit the chips very close together) gets its answer here in TSVs, interposers and stack bonding. **The memory problem and the packaging problem were the same problem.**

PRIMARY SOURCES

SK hynix Newsroom, "Semiconductor Back-End Process Episode 6: The Eight Steps of Assembling Conventional Packages" · ScienceON, "Artificial Intelligence Semiconductor and Packaging Technology Trend" (2023, in Korean) · ScienceON, "Advances and Research Trends in 3D Packaging-Based High-Bandwidth and Large-Capacity Memory Technologies for Next-Generation AI Hardware" (2026, in Korean) · Siemens EDA, "Comprehensive 3D IC guide to heterogeneous integration" · TSMC 3DFabric (CoWoS, InFO, SoIC) · JEDEC JESD47 standard page · SK hynix, 12-high HBM4E samples shipped (2026-06-18) · UCIe specifications page

Learning objectives

1. Explain how the purpose of traditional back-end processing differs from that of advanced packaging.
2. Calculate the gains and losses of chiplet partitioning with a yield model.
3. Explain which physical problem each of TSVs, interposers and hybrid bonding solves.
4. Tell 2.5D and 3D apart precisely, and choose the approach that fits a design goal.
5. Explain why CoWoS capacity is the bottleneck of AI chip supply.

0. Self-study guide — for readers working alone

> **THIS SESSION IS WHERE THE PREVIOUS FIVE SESSIONS MEET**
>
> Bandwidth from Session 2, HBM from Session 3, yield from Session 4 and the reticle limit from Session 5 merge here into a single problem: **"If you cannot make a big chip but need big performance, what do you do?"** Exercise 1 (optimal chiplet partitioning) gives the answer in numbers. Four of this session's five exercises reuse tools from earlier sessions, so if you understood those, they go surprisingly fast.

What this session brings over from earlier ones — it leans on them heavily

FROM	WHAT
Session 4	$Y = e^{-AD_0}$ and the GDPW formula — **used as is in Exercise 1**
Session 5 Exercise 5	Reticle limit 858 mm², 429 mm² for High-NA
Session 3 Exercise 2	pJ/bit — **used as is in Exercise 2**
Session 3	HBM stack structure, TSVs, stack yield y^n, the stacking methods in volume production (TSV + microbumps, MR-MUF and TC-NCF) and the next-generation candidate, hybrid bonding
Session 2	Why bandwidth is everything

> If you have not done Exercise 4 (chiplet gains and losses) of Session 4, **go back and do it now.** Exercise 1 of this session generalizes it.

Prerequisite check

CONCEPT	ONE-LINE ANSWER
Parasitic capacitance	The electrical capacitance a wire has without meaning to. Roughly proportional to length; charging and discharging it costs energy
$E = \frac{1}{2}CV^2$	The energy needed to charge a capacitance C to a voltage V
Thermal resistance $R = t\ /\ (kA)$	The resistance heat meets when passing through a layer of thickness t, thermal conductivity k and area A [K/W]. θJA and θJC, which roll a whole package into one value, come in section 7
Pitch	The center-to-center distance between neighboring wires or bumps
Signal integrity	Whether a transmitted signal arrives without distortion or noise

Reading order (130 minutes in total recommended)

STEP	SECTIONS	TIME	NOTE
1	Sections 1–3 (overview of the back end, why split)	25 min	An extension of Session 4
2	Sections 4–6 (2.5D and 3D, TSVs, bonding)	35 min	The HBM structure is completed here
3	**Exercise 1**	**25 min**	**The heart of this session.** Fill in the table yourself
4	Section 7 on heat and reliability + Exercise 4	20 min	The bridge to Session 7
5	Sections 8 and 9 on UCIe and signal integrity + Exercises 2, 3 and 5	25 min	

The one sentence that runs through this session

> **Packaging is not "a box to put the chip in" but "an extension of the wiring."** How close you place the dies decides bandwidth, power and heat all at once. Exercise 2 shows this with a number: a factor of 200.

A common sticking point — when 2.5D and 3D get confusing

The terms are confusing because the names refer not to geometry but to **the way things are connected.**

	PLACEMENT	CONNECTION	TYPICAL EXAMPLE
2D	Side by side, on a substrate	Substrate wiring (long)	Traditional multichip module
2.5D	Side by side, **on an interposer**	Interposer wiring (short)	GPU + HBM (CoWoS)
3D	**Stacked one on top of another**	TSV + microbumps, or hybrid bonding (shortest)	Inside an HBM stack (microbumps today), logic on logic

To remember it: 2.5D is "beside, but very close"; 3D is "on top." And **today's AI accelerators use both at once** — inside the HBM stack it is 3D, between the GPU and the HBM it is 2.5D.

---

1. Getting the intuition — the purpose of the back end has changed

The back end of old

```
Wafer test (electrical test → wafer burn-in → functional test → repair → retest)
  → backgrinding (thinning the back) → dicing (cutting) → die attach → wire bonding
  → molding (resin encapsulation) → ball attach and singulation (for BGA) → package test →
shipment
```

Testing is not one step at the very end. **Chips are screened first in wafer form, before**

cutting, and inspected again after they are packaged. The earlier testing produces the KGD of section 2.

A package does four jobs: ① **it protects the chip physically and chemically,** ② **it distributes power to the chip,** ③ **it lets signals pass to and from the outside world, and** ④ **it removes the heat the chip produces.** The *Semiconductor Packaging* volume of the second edition of the *Microelectronics Packaging Handbook* edited by Tummala and others (Chapman & Hall, 1997), counted as the standard reference in this field, says the same in its publisher's description: "Electronic packaging is defined as interconnection, powering, cooling, and protecting semiconductor chips for reliable systems." (The bibliographic details were confirmed from the publisher's catalog; the text of the book itself was not checked.) The fourth job is the subject of section 7 of this session and of Session 7. In the old days this work was relatively simple, so it was outsourced to OSATs in regions with cheap labor, and it added little value.

Packages are also divided into levels: the die itself, the single package that holds the die (first-level package), the board the package is mounted on (second level), and the system that gathers the boards. Whether the die counts as level zero, and what number the system gets, varies from text to text. Look less at the number and more at **which level connects to which.** Advanced packaging puts several dies and an interposer inside the first-level package, pulling connections that the board used to make into the package.

Advanced packaging today

The purpose is entirely different: **to connect several chips to each other at ultra-high bandwidth so that they behave like one giant chip.**

Packaging now decides performance. With the same GPU die, system performance varies widely with how many HBM stacks are attached and what kind of interposer is used. So **foundries have begun to absorb packaging themselves** (TSMC's CoWoS and SoIC, Intel's Foveros and EMIB, Samsung's I-Cube and X-Cube).

> **In one sentence:** if the front end is "the race to make transistors small," the back end is "the race to put chips close together." And right now the second race is paying off faster.

---

2. Why the center of gravity moved — three pressures

Pressure ① Yield — a large die cannot be made

Write out the formula from Session 4 again: $Y = e^{-AD_0}$.

Comparing the calculations (assuming $D_0 = 0.1/\text{cm}^2$)

APPROACH	DIE CONFIGURATION	INDIVIDUAL YIELD	SYSTEM YIELD
Single large die	800 mm² × 1	44.9%	**44.9%**
Chiplet partition	200 mm² × 4	81.9% (each)	$0.819^4 = 45.0\%$ ← huh?

Wait — aren't they the same? Calculated naively, they are. In e^{-AD_0}, even if you divide the area into four, you need all four, so the yields multiply. Converting the area to cm², $\left(e^{-2\,\mathrm{cm}^2 \times D_0}\right)^4 = e^{-8\,\mathrm{cm}^2 \times D_0} = e^{-0.8} = 44.9\%$, the same (800 mm² = 8 cm²).

This is where it starts to matter. Chiplets win for the following reasons.

1. **KGD screening becomes possible.** Small dies are easy and cheap to test individually. Assemble only good ones, and system yield approaches assembly yield. A large die with a defect is thrown away whole.
2. **What you throw away differs.** If an 800mm² die is defective, you discard 800mm². If one 200mm² chiplet is defective, you discard only 200mm², and the other three go into inventory.
3. **They are reused.** The same I/O chiplet can serve several products. NRE is spread out.
4. **Mixed nodes become possible.** See pressure ③.
5. **More dies fit on a wafer, so the share lost at the edge goes down.**

Where does KGD come from? From wafer test, at the front of the flow chart in section 1. In wafer form the chips' electrical characteristics are measured, high temperature and voltage are applied to expose in advance the chips that would die early (burn-in), and their functions are checked. Memory does not stop there. Rows and columns with bad cells are swapped for spare rows and columns (repair), and the chips are tested again. Standards give a clue to how long early failures must be screened. JEDEC JESD47I (2012) sets the qualification test that measures the early life failure rate (ELFR) at a junction temperature of 125 °C or higher for 48–168 hours. That is a condition of product qualification testing, not the volume-production burn-in condition itself.

The exact comparison formula: for n chiplets, each of area A / n, with assembly yield y_{asm} after KGD screening,

$$Y_{\text{chiplet-system}} \approx y_{\text{asm}}, \qquad \text{cost per good system} \approx \frac{1}{y_{\text{asm}}}\left[n \times \frac{c_{\text{wafer}}}{\text{GDPW}(A / n) \times e^{-(A / n)D_0}} + c_{\text{assembly}}\right]$$

The bracket is the cost of one assembly attempt. The good systems that survive carry the cost of the systems lost in assembly, hence the division by y_{asm}. The solution to Exercise 1 sets c_{assembly} to 0 and calculates with this formula.

The price: the interface circuits (PHY) for chiplet-to-chiplet communication eat area and power, and communication across dies is slower and more expensive than on-die communication. **Partitioning does not always win** — Exercise 4 of Session 4 computed the break-even point.

Pressure ② The exposure field limit — it physically cannot be made

A lithography tool prints only a fixed area at a time (the reticle field). ASML gives the maximum exposure field of its 0.33 NA EUV tools (NXE) as 26 mm × 33 mm. That works out to 858 mm^2. **ASML states: "Because of their anamorphic optics, EXE systems have exposure fields half the size of their NXE predecessors." That works out to about 429 mm^2.** Why it is half was calculated in Session 5.

A chip larger than this **cannot, in principle, be made in one shot.** Stitching (joining fields together) exists, but overlay errors and yield problems keep it from being common.

> ★ **The link foreshadowed in Session 5:** adopting High-NA EUV gains resolution but loses field. So **a design that needs a die larger than 429 mm^2 must, on moving to High-NA, choose between stitching and chiplets, and the incentive to choose chiplets grows.** If the required die area fits within 429 mm^2, if scaling squeezes the same circuit into it, or if you accept the alignment and yield burden of stitching, you need not partition. This link, in which the next step of scaling raises the incentive for a packaging-side solution, ties Session 5 to this session.

Pressure ③ Heterogeneous optimization — not every circuit benefits from scaling

CIRCUIT TYPE	GAIN FROM SCALING	BEST CHOICE
Logic (compute)	Large	Latest node
SRAM (cache)	**Shrinking** — SRAM cell area now scales much more slowly than logic	Limited gain from the latest node
Analog, I/O	**Almost none** — needs voltage swing and physical size	Mature nodes are far cheaper
Power management	None	Mature node

Wafers at the latest node cost far more than at mature nodes. **Putting circuits that gain nothing from scaling on the latest node is waste.** So logic is made at the latest node and I/O at a mature node, and the two are joined. AMD demonstrated this strategy at scale, and it has now become standard in the industry.

---

3. Dicing — even cutting has become hard

METHOD	PRINCIPLE	CHARACTERISTICS
Blade	Cutting with a diamond blade	Traditional, cheap. Chipping (broken edges), kerf (cut width) loss
Laser scribing	Cut a groove with a laser, then separate	Small kerf, heat-affected zone
Stealth dicing	Focus a laser **inside** the silicon to form a modified layer, then split by expanding the tape	Almost no surface damage or particles, **suited to thin wafers**
Plasma dicing	Separation by etching (combined with DBG)	Any shape possible, no chipping, excellent die strength

Why it has become hard

1. **Wafers have become thin.** Wafers for TSVs and stacking are ground down to a few tens of μm. Something as thin as paper has to be cut and handled without breaking.
2. **Kerf has to shrink.** The cut width is area thrown away. The smaller the die, the larger the share of kerf.
3. **Die edge strength has become important.** Stacking applies stress, so a microcrack on the cut face leads to failure later.
4. **Low-k dielectrics are weak.** The low-permittivity materials in the wiring layers are mechanically fragile and delaminate under blade cutting.

---

4. TSV — vertical wiring that runs through the silicon

Why it is needed

The problem seen in Session 3: **to get bandwidth you must pull out a great many wires, and to do that the chips must be very close.** Stacking chips vertically shrinks the distance to the μm scale, but there has to be a passage connecting the upper and lower chips. That passage is the TSV.

Why it is better electrically

Wire delay is roughly proportional to RC, and drive energy is proportional to $E = \frac{1}{2}CV^2$. A shorter wire reduces both R and C.

- Wiring that detours through the package substrate: **a few mm to a few tens of mm**
- Interposer wiring: **a few hundred μm to a few mm**
- TSV (through the thickness of the chip): **a few tens of μm**

When the length drops 100-fold, parasitic capacitance drops by roughly as much, and the energy to transmit each bit falls sharply. This is the core of why HBM's pJ/bit is much better

than GDDR's.

Process flow

```
1. High-aspect-ratio etching (Bosch process, etc.) — a deep, narrow hole
2. Insulator deposition (ALD/CVD) — electrical isolation from the silicon
3. Barrier/seed layer deposition (ALD/PVD) — blocks Cu diffusion + seed for plating
4. Copper electroplating — fill without voids (bottom-up fill)
5. CMP — remove excess copper from the surface
6. Wafer thinning (backgrinding) — grind the back to expose the TSVs
7. Backside processing — insulation and pad formation
```

Hard problems

- **Void-free filling:** the deep hole must be filled with copper without gaps. If the mouth closes first, a void forms inside. Plating chemistry that uses additives to fill from the bottom up is the key technology.
- **Thermal expansion mismatch (CTE mismatch):** copper and silicon have different coefficients of thermal expansion. When the temperature changes, stress builds around the TSV, and that stress shifts the characteristics of nearby transistors. So a **keep-out zone** is set around each TSV where no devices are placed. **This is where area overhead arises.**
- **When to drill:** drilling before the transistors are made is via-first; after the transistors but before the wiring layers are built, via-middle; after the wiring layers are finished, via-last. The order changes which high-temperature steps the TSV must survive and how the keep-out zone is designed. Stacked memory such as HBM is cited as a typical example on the via-first and via-middle side.
- **Handling thinned wafers (WSS, wafer support system):** a wafer a few tens of μm thick cannot hold its shape and warps. So it is temporarily attached to a rigid carrier while its back side is processed. In order: before attaching it to the carrier, the edge of the wafer's front side is trimmed away (edge trimming). The wafer edge is rounded, so if it were attached as is and ground to a few tens of μm, the edge would be left as thin as a knife blade, and the later steps would break it from that tip. Next the wafer is attached to the carrier with a temporary adhesive, the back is ground to expose the TSVs, backside bumps are made, and the carrier is removed (debonding). Debonding must not break the thinned wafer, must not damage the bumps and must leave no adhesive behind, so the process conditions are demanding. Debonding methods can be classified by what neutralizes the adhesive layer: mechanical methods that lift the carrier off by force, thermal methods that soften the adhesive with heat, chemical methods that dissolve it with a solvent, and optical methods that decompose the adhesive layer with a laser. The laser method needs light to pass through the carrier, so it uses a glass carrier. (The process order and debonding methods are summarized from episode 8 of the SK hynix Newsroom series "Semiconductor Back-End Process.")
- **Tested and shipped as a stack:** HBM is made by stacking a base die and DRAM dies, molding them, testing them as a stack and shipping them as a KGSD (known good stacked die). What the maker of the GPU package receives is not individual DRAM dies but a verified stack.

TGV and glass substrates

TGV (through-glass via): a via that passes through glass. Why glass substrates draw attention:

ADVANTAGE OF GLASS	WHAT IT MEANS
Low dielectric loss	Less loss of high-frequency signals → good for high-speed interconnect
Excellent flatness and stiffness	Does not warp even over a large area → good for large packages
Adjustable CTE	Can be matched to silicon → less stress
Large-panel processing possible	Processed as square panels rather than round wafers → productivity

Remaining challenges: glass is brittle, so cracks propagate easily. Via formation (laser-induced etching and the like), metallization and breakage during handling are the gates to practical use. The schedule is still at the target stage. Announcing its glass substrates in September 2023, Intel said it "is on track to deliver complete glass substrate solutions to the market in the second half of this decade," and in July 2026 it announced a collaboration with the glass-processing company Lens Technology to accelerate development of glass substrate packaging. As of October 2026 we found no official announcement of volume production of glass-substrate packages.

---

5. Interposers — the board that connects the chips

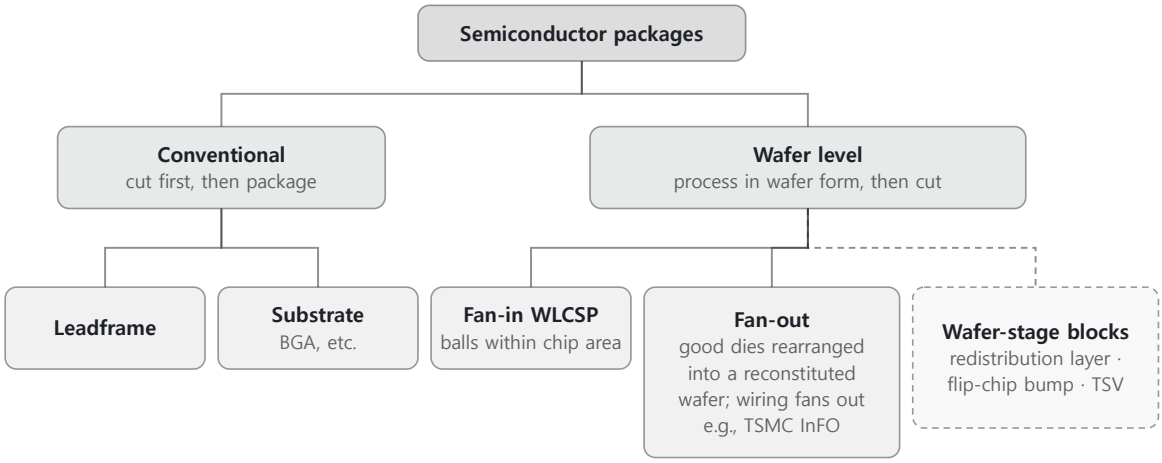
Types and trade-offs

TYPE	WIRING DENSITY	COST	USE
Organic substrate	Low (wiring at a pitch of a few µm is hard)	Low	Ordinary packages, low-bandwidth connections
RDL interposer (fan-out)	Medium	Medium	InFO and others. Redistribution layers formed on organic material
Silicon interposer	**High** (fine damascene wiring possible)	**High**	**The standard for HBM connection.** CoWoS-S
Silicon bridge (EMIB, etc.)	High (only where needed)	Medium	Instead of making the whole interposer silicon, a piece of silicon is inserted only at the connection points
Glass interposer	High (potentially)	?	Under development

Package classification — fan-in and fan-out

To see where the "RDL interposer (fan-out)" in the table above comes from, you first need the classification of packages. The big division is **when the wafer is cut.** A conventional package first cuts the wafer, attaches each die to a leadframe or a substrate, and encapsulates it. A wafer-level package finishes package steps such as redistribution layers and bumps while still in wafer form, and only then cuts.

Within wafer level there is a further split. In **fan-in (WLCSP)**, the solder balls sit only within the chip's area. If the chip is small, the number of balls is tied to its area. In **fan-out**, only good dies are picked, laid out again with spacing between them, and the gaps filled with molding compound (EMC) to make a "reconstituted wafer"; the redistribution layers are then spread out over the molding beyond the chip, and the balls placed there. Picking only good dies to lay out shares its root with the KGD logic of section 2. TSMC's InFO (Integrated Fan-Out) is a wafer-level package of this family, and TSMC states that it "has been shipping InFO in high volume since 2016." InFO also has a variant that ties a logic die and HBM together with redistribution layers (InFO_MS), which TSMC presents as offering "a cost/performance alternative to CoWoS for InFO size < 1X reticle."



Two criteria divide the tree: "when is the wafer cut?" and "do the balls go beyond the chip's area?" Redistribution layers, bumps and TSVs are less package types than wafer-stage building blocks that many packages share, so they are grouped separately with a dashed line.

Why HBM needs a silicon interposer

One HBM4 stack is 2,048 bits wide. Eight stacks make **16,384 data wires**, with control, power and ground on top. These wires have to be packed densely across a distance of a few mm from the edge of the GPU die to the HBM.

That density is impossible at the wiring pitch of an organic substrate. **A silicon interposer is made with semiconductor processes (damascene wiring), so pitches below a μm are possible.** In other words, an interposer is "a chip that is nothing but wiring."

The problem this creates: a silicon interposer is also made with semiconductor processes, so **it is subject to the exposure field limit.** To carry a GPU and eight HBM stacks, the interposer has to be larger than the reticle. Hence:

- **CoWoS-S:** a silicon interposer. Stitching is used to make it several times the reticle size. TSMC says it has been in production since 2012, with the interposer reaching up to 3.3 times the reticle (about 2,700 mm^2 in TSMC's figure; multiplying this book's 858 mm^2 gives about 2,830 mm^2)
- **CoWoS-R:** based on RDL (redistribution layers). Can be made larger and is cheaper, but the density is lower. TSMC recommends CoWoS-L or CoWoS-R for interposers larger than

3.3 times the reticle and says -R has been in volume production since 2023

- **CoWoS-L:** local silicon interconnect (LSI) bridges inserted into the RDL. **Silicon only where high density is needed, RDL elsewhere** — combining the strengths of the two. The mainstream for the latest large AI packages. TSMC's "first CoWoS®-L at 3.5X-reticle size has been in volume production since 2024" (TSMC, checked October 2026)

Why CoWoS is the bottleneck of AI chip supply

1. **The process is complex and slow.** It is multistage — interposer fabrication + chip bonding + substrate assembly — with yield loss at every stage.
2. **Adding capacity takes time.** Ordering, installing and qualifying equipment takes years.
3. **Demand exploded.** Each AI accelerator needs a very large interposer area, and shipments surged.
4. **It is effectively concentrated in one particular foundry.**

On its CoWoS page TSMC writes: "With the advent of generative AI in late 2022, market demand for CoWoS® solutions has become even greater," and it "continues to expand its advanced packaging capacity" (checked October 2026). The size of that capacity and the allocations by company vary widely from report to report, so this book does not state them as figures.

Why it is not easy to reroute — the assembly order. There are two broad orders for assembling a 2.5D package. **CoW-first** keeps the interposer in wafer form, attaches the chips onto it first, molds at the wafer level, then attaches the wafer to a temporary carrier, grinds the back to expose the TSVs, builds the backside redistribution layers and bumps, and finally cuts the wafer and mounts the pieces on substrates. There is a process report that actually went through this order with a 300 mm silicon interposer (Ho et al., EPTC 2014). The name CoWoS itself (chip-on-wafer-on-substrate) points to this order. **CoS-first** cuts the interposer first, attaches it to the organic substrate, and places the chips on top. What separates the two is warpage. Silicon's coefficient of thermal expansion is much smaller than an organic substrate's. If a large silicon interposer is attached to the substrate first, the two layers shrink differently as they cool from the bonding temperature, and warp; when the chips are then placed, tens of thousands of microbumps have to be aligned on that warped surface. CoW-first finishes the fine bonding first on a flat, rigid wafer, which reduces the problem (the warpage of the molded interposer still has to be managed in CoW-first). The larger the interposer, the wider the gap. But CoW-first needs wafer-level molding, thinning and redistribution steps, so places without such equipment and experience cannot easily take over. That is one of the reasons the bottleneck gathers in one place.

> **The "bottleneck keeps moving" of Session 1 takes concrete form here.** Even with room to spare in logic wafer capacity, if CoWoS capacity and HBM supply fall short, only that many AI accelerators can be made. **What caps AI chip shipments is not the latest node but packaging and memory.**

---

6. Bonding — a generational change in how chips are joined

The path of evolution

GENERATION	METHOD	PITCH	CHARACTERISTICS
Wire bonding	Gold or copper wires connect chip pads to the substrate	Tens of µm	Connections only at the edge, limited I/O count, high inductance
Flip chip (C4 bumps)	The chip is flipped and joined directly with solder bumps	~100 µm	**The whole face can be used** → I/O count soars
Microbumps	Small solder bumps	~40 µm or less	The workhorse of 2.5D/3D stacking
Microbumps + gap fill (MR-MUF / TC-NCF)	MR-MUF: stack all the chips, reflow them all at once to join the bumps, then flow a liquid protective material between the chips and cure it. TC-NCF: lay a non-conductive film on each layer and press the layers on one at a time with heat and pressure	Same as microbumps	**The stacking method of volume-production HBM as of October 2026.** The bumps are the same; what differs is how the gap is filled
Hybrid bonding (Cu-Cu)	**Copper pads bonded directly to each other without bumps** + the surrounding dielectric bonded too	**Sub-10 µm**	Highest density, lowest resistance, lowest inductance

Stack form and yield — W2W and D2W. Dies are stacked as chip on chip (C2C), die on wafer (D2W, also called C2W) or wafer on wafer (W2W). W2W joins two wafers in one go and is highly productive, but the upper and lower dies must have the same size and layout, and above all **you cannot pick good dies to join.** A defective die on the upper wafer lands as is on a good die of the lower wafer. So when n layers are stacked, the stack-yield formula of section 5 of Session 3 applies directly.

$$Y_{\mathrm{W2W}} = y^{n} \times y_{\mathrm{bond}}^{\,n-1}$$

D2W cuts and tests the dies, then picks only the good ones (KGD) and joins them onto the wafer. Suppose the bottom wafer is also tested in advance and stacking happens only on its good die sites. If screening is perfect, the die-yield term drops out and only the bonding yield remains.

$$Y_{\mathrm{D2W}} \approx y_{\mathrm{bond}}^{\,n-1}$$

The denominator of this formula is the good sites of the bottom wafer. If you also stack on the bottom wafer's defective sites, or count those sites in the denominator, the bottom die yield is multiplied in once and the result becomes $y \cdot y_{\mathrm{bond}}^{\,n-1}$. With the assumed values below and $n = 8$, that is 0.869, which is 2.09 times W2W rather than 2.32 times. The value of the discarded bottom sites remains as a cost.

With die yield $y = 0.9$ and the yield of one bonding step $y_{\mathrm{bond}} = 0.995$ (both assumed values; the bonding yield is the same as the example in section 5 of Session 3), the calculation goes as follows.

STACK COUNT n	W2W $y^n y_{\mathrm{BOND}}^{n-1}$	D2W y_{BOND}^{n-1}	D2W ÷ W2W
2	0.806	0.995	1.23 times
8	0.416	0.966	2.32 times
12	0.267	0.946	3.54 times
16	0.172	0.928	5.40 times

With two layers the gap is in the 20% range, but as layers are added it widens several-fold. That is why W2W suits processes with few layers, dies of the same size and high yield. TSMC, too, presents its wafer-on-wafer SoIC-X as "suitable for high yielding nodes and the same die size applications or design." Even in D2W, though, the value of the discarded defective dies remains as a cost. The yield loss does not disappear; **it moves to before stacking, so good dies are no longer thrown away along with bad ones.** It is the KGD logic of pressure ① in section 2, extended in the stacking direction.

The figure below shows eight layers stacked with the assumed values of the table above ($y = 0.9$, $y_{\mathrm{bond}} = 0.995$). In the screen edition you can change the die yield, the bonding yield and the stack count.



Figure 6-1. Stacking 8 layers with die yield 0.9 and bonding yield 0.995 (assumed) gives a W2W yield of 0.416, while D2W, which bonds only good dies, gives 0.966: a 2.32-fold difference. — In the HTML edition you can change the values here (HTML edition, Session 6, Section 6).

Why hybrid bonding matters

The problem with bumps: bumps have a physical size. To shrink the pitch you have to make the bumps smaller, but small solder joints are less reliable and have higher electrical resistance. The underfill material that fills the space between bumps also becomes hard to inject as the space narrows. **Bump-based methods hit a wall at a pitch of roughly a few tens of µm.**

How hybrid bonding works: the surfaces of two wafers (or dies) are made flat at the atomic level, and surfaces where the copper pads and the dielectric (mainly SiO_2) are exposed together are pressed face to face. At room temperature the dielectric surfaces bond first (van der Waals → covalent bonds); a later anneal makes the copper expand and diffusion-bond to the copper opposite.

What you gain: the pitch drops into the sub-10 µm range, so the number of connections per unit area exceeds that of microbumps (about 40 µm) by more than 16 times (100 times at a 4 µm pitch), the resistance and parasitics of the joint fall dramatically, and the stack becomes lower.

What it demands (extreme)

- **Surface flatness:** on the scale of angstroms. CMP quality decides everything
- **Cleanliness:** a single particle ruins the bond at that spot
- **Alignment accuracy:** a few hundred nm or less. Die-to-die alignment equipment becomes a new bottleneck
- **Copper recess control:** the copper must sit slightly below the dielectric, and that amount has to be controlled at the nanometer level

Where it is used: CMOS image sensors have been mass-produced this way for many years. 3D logic stacking is moving in this direction too (TSMC's SoIC and others; TSMC states that SoIC's "bond pitch starts from the sub-10µm rule" and that "its 3nm chip stacking technology successfully entered volume production in 2025"). For HBM it is **a candidate for next-generation high stacks,** because as the layer count rises, the height of the bumps and gap-fill layers runs into the stack height limit. SK hynix said in 2023 that in 2022 it had stacked an 8-high HBM2E with hybrid bonding and completed electrical tests, but the 12-high HBM4E samples it supplied to customers in June 2026 used Advanced MR-MUF. As of October 2026 we found no manufacturer announcement of volume production of HBM using hybrid bonding (the same judgment as in section 4 of Session 3).

> **A point to note:** hybrid bonding **demands front-end precision of the back end.** This is why foundries, not OSATs, are taking over this area.

7. 2.5D and 3D — the exact distinction

Definitions

- **2.5D:** chips placed **side by side (horizontally)** on an interposer. The chips do not overlap vertically. **Typical: GPU + HBM stacks (CoWoS)**
- **3D:** chips **overlapped vertically.** Typical: the DRAM stacking inside an HBM stack, logic or cache stacked on logic (SoIC, Foveros)

Where the name "2.5D" comes from: not fully 3D, but more integrated than a flat layout. The HBM stack itself is 3D, so a GPU+HBM system is a structure with **3D inside 2.5D.**

Comparison

	2.5D	3D
Connection distance	mm scale (via the interposer)	µm scale (direct stacking)
Bandwidth density	High	**Very high**
Heat removal	Each chip can see the heat sink	**The upper chip covers the lower one and traps the heat**
Area	Large interposer area	Small footprint
Process difficulty	Medium	High
Design complexity	Medium	High (heat, power and EDA alike)

The greatest enemy of 3D — heat

This is the fundamental hard problem of 3D stacking.

- Overlapping chips multiplies the heat per unit area by the number of layers.
- **The heat of the lower chip has to pass through the upper chip to get out,** and the silicon and bonding layers have thermal resistance. The chip on top effectively acts as a blanket.
- As temperature rises, leakage current grows, and as leakage grows, more heat is produced: a **positive feedback** loop.
- So 3D logic stacking usually **puts the layer that generates less heat on top** (for example, cache SRAM on logic). Overlapping high-power logic with high-power logic is still hard.

The same problem in HBM: in a 12–16-high stack, the heat of the bottom die does not escape well. Together with the stack height limit, this constrains the number of layers. Manufacturers advertise this value too. When SK hynix supplied 12-high HBM4E samples in June 2026, it said it had lowered thermal resistance by about 17% compared with HBM4.

Heat leaving the package — θ and ψ. The $R = t / (kA)$ seen so far is the thermal resistance of a single layer. Datasheets roll the whole package into a few numbers. **θJA** (junction to ambient air), **θJC** (junction to case) and **θJB** (junction to board), defined in the JEDEC JESD51 family, are the difference between the junction temperature T_j and the temperature at a reference point, divided by the power dissipated P. For example, since $\theta_{JA} = (T_j - T_a) / P$, it follows that $T_j = T_a + P\,\theta_{JA}$. But θJA is measured on a standard test board in still air or at a

fixed air speed, so the value changes when the board and airflow change. A Renesas technical brief notes that a test board with two copper planes (1S2P) lowers θJA by roughly 20–40% or more compared with a board with none (1S0P), and TI notes that θJA is a value for comparing packages under the same conditions, not for calculating T_j in an actual design. To work backward to T_j from the temperature of the top of the package or of the board on a real board, you use **ψJT and ψJB.** The formula looks the same as θ's, but since not all the heat flows between those two points, it is not a true thermal resistance but a characterization parameter.

Seeing it as a series chain. In a package with a heat sink attached, heat flows in turn from the junction → case → TIM → heat sink (or cold plate) → ambient (air or coolant). Being in series, the resistances add.

$$T_j = T_a + P\,(\theta_{JC} + \theta_{\mathrm{TIM}} + \theta_{SA})$$

θ_{SA} is the thermal resistance from the heat sink to ambient. Take one worked example. The values below are all **assumed values,** not those of any particular product. With P = 1,000 W, T_a = 35 °C (you may read it as the coolant inlet temperature), θ_{JC} = 0.015 K/W, θ_{TIM} = 0.025 K/W and θ_{SA} = 0.020 K/W, the sum is 0.060 K/W. The temperature differences at each stage are 15 K, 25 K and 20 K, 60 K in total, so T_j = 95 °C. This has to be below the maximum junction temperature $T_{j,\ \mathrm{max}}$ that each product's datasheet sets. The assumed values give the largest share to the TIM to preview the point of section 3 of Session 7: the thin layer between chip and cold plate easily becomes the bottleneck of the whole chain.

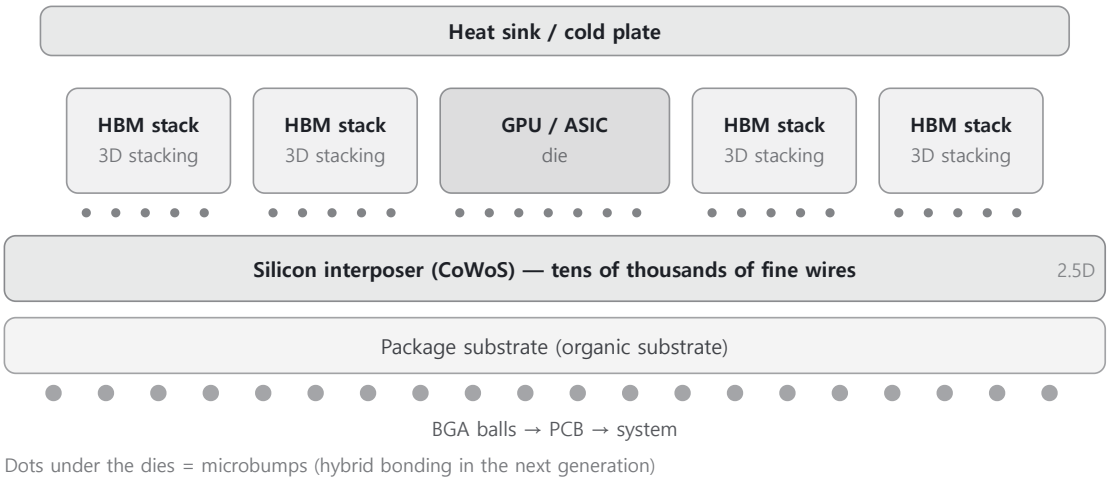
> **On to Session 7:** the data center cooling problem is the heat problem extended from inside the chip to the scale of the rack.

The figure below shows how temperature builds up from ambient to junction with the assumed values of the worked example above. In the screen edition you can change the power dissipated, the ambient temperature and the three thermal resistances, and enter a datasheet's maximum junction temperature to see whether it is exceeded.



Figure 6-2. Removing 1,000 W from a chip to 35 °C ambient through a chain of θJC 0.015, θTIM 0.025 and θSA 0.020 K/W (assumed) gives stage temperature differences of 15, 25 and 20 K and a junction temperature of 95 °C. — In the HTML edition you can change the values here (HTML edition, Session 6, Section 7).

The structure of a real AI accelerator package



Dots under the dies = microbumps (hybrid bonding in the next generation)

What to check here

- Inside the HBM stack = 3D; the relation between GPU and HBM = 2.5D
- The whole package area is several times the reticle → extension techniques such as CoWoS-L are needed
- One such package draws power in the kilowatt range, and that heat has to leave through the cold plate (Session 7)

Package reliability — how much heat and moisture can it take?

Heat does not stop at raising the temperature. Long exposure to heat degrades materials, and each power cycle makes materials expand and contract by different amounts, fatiguing the joints. Moisture brings corrosion. So new products go through qualification tests that deliberately break them under harsh conditions before shipment. Below are the main items taken from Table 1 (device tests) and Table 2 (tests for nonhermetic packages) of JEDEC JESD47I (July 2012 edition). JEDEC issued JESD47N in August 2026. We did not check against the text of the latest edition.

TEST	STANDARD	CONDITIONS (JESD47I)	SAMPLE AND PASS CRITERION	FAILURE TARGETED
HTOL (high temperature operating life)	JESD22-A108	Junction temperature ≥ 125 °C, ≥ maximum supply voltage, 1,000 hours	3 lots × 77 units, 0 fails	Failures from heat and voltage during operation
ELFR (early life failure rate)	JESD22-A108, JESD74	Junction temperature ≥ 125 °C, 48–168 hours	ELFR sample table	Early failures in the first months after shipment
HTSL (high temperature storage life)	JESD22-A103	150 °C, 1,000 hours	3 lots × 25 units, 0 fails	Material degradation such as diffusion, oxidation and intermetallic growth
THB (temperature humidity bias)	JESD22-A101	85 °C, 85% relative humidity, bias applied, 1,000 hours	3 lots × 25 units, 0 fails	Corrosion and ion migration from moisture
HAST (highly accelerated stress test)	JESD22-A110	130 °C or 110 °C, 85% relative humidity, bias applied, 96 hours or 264 hours	3 lots × 25 units, 0 fails	A shortened, accelerated THB
UHAST (unbiased HAST)	JESD22-A118	96 hours at 130 °C and 85%, or 264 hours at 110 °C and 85%	3 lots × 25 units, 0 fails	Corrosion
TC (temperature cycling)	JESD22-A104	e.g., condition B is −55 to +125 °C, 700 cycles	3 lots × 25 units, 0 fails	Thermomechanical damage from thermal expansion differences

Source: selected from Tables 1 and 2 of JEDEC JESD47I (2012-07). The layout of the table and the "Failure targeted" column are this book's own summary.

There are rules outside the table, too. THB, HAST, TC and UHAST (and autoclave) are preceded by **preconditioning:** following JESD22-A113, the parts are soaked in moisture

according to their moisture sensitivity level (MSL) under J-STD-020 and put through reflow, giving them in advance the stress of being soldered to a board. Autoclave, which steams parts in saturated vapor, can cause damage that does not occur in real use, so it is not recommended as a qualification test, and HAST is required instead for organic-substrate packages. There is also a note that the TC conditions that go below freezing, G, B and C, may not suit organic-substrate flip-chip packages. **The thermal expansion mismatch and warpage that trouble 3D stacks and large interposers are exactly the failures TC targets.** The keep-out zone of section 4 and CoW-first in section 5 both came from the same physics.

How many years is 1,000 hours? — the Arrhenius conversion. For failures accelerated by temperature, test time is commonly converted to use time with the Arrhenius equation.

$$AF = \exp\left[\frac{E_a}{k}\left(\frac{1}{T_{\text{use}}} - \frac{1}{T_{\text{stress}}}\right)\right]$$

E_a is the apparent activation energy, $k = 8.617 \times 10^{-5}$ eV/K is Boltzmann's constant, and temperatures are absolute (K). With a test at 125 °C (398.15 K), use at 55 °C (328.15 K) and E_a = 0.7 eV, $AF \approx 77.7$. HTOL's 1,000 hours correspond to about 77,700 hours; dividing by 8,760 hours a year gives **about 8.87 years.** JESD47I, in a note to Table 1, gives the same conditions as an example with an acceleration factor of 78.6, about 9 years (the last digit differs because of the constant and rounding).

Change only E_a under the same conditions and the conclusion swings widely.

E_a	ACCELERATION FACTOR AF	EQUIVALENT USE PERIOD OF 1,000 HOURS OF HTOL
0.5 eV	22.4	about 2.56 years
0.7 eV	77.7	about 8.87 years
0.9 eV	269.3	about 30.7 years

Between 0.5 eV and 0.9 eV, the equivalent use period of the same test differs about 12-fold. E_a differs by failure mechanism. That is why one test cannot tell you "this chip will last nine years." JESD47I itself adds the caveat that this period may be shorter than actual use requirements and depends on the failure mechanism and use environment, and says long-term wear-out mechanisms should be covered by wafer-level reliability data. JEDEC JEP122 collects acceleration models by failure mechanism.

The figure below is for E_a = 0.7 eV, 125 °C → 55 °C, 1,000 hours. In the screen edition you can change E_a, the test and use temperatures, and the test time.

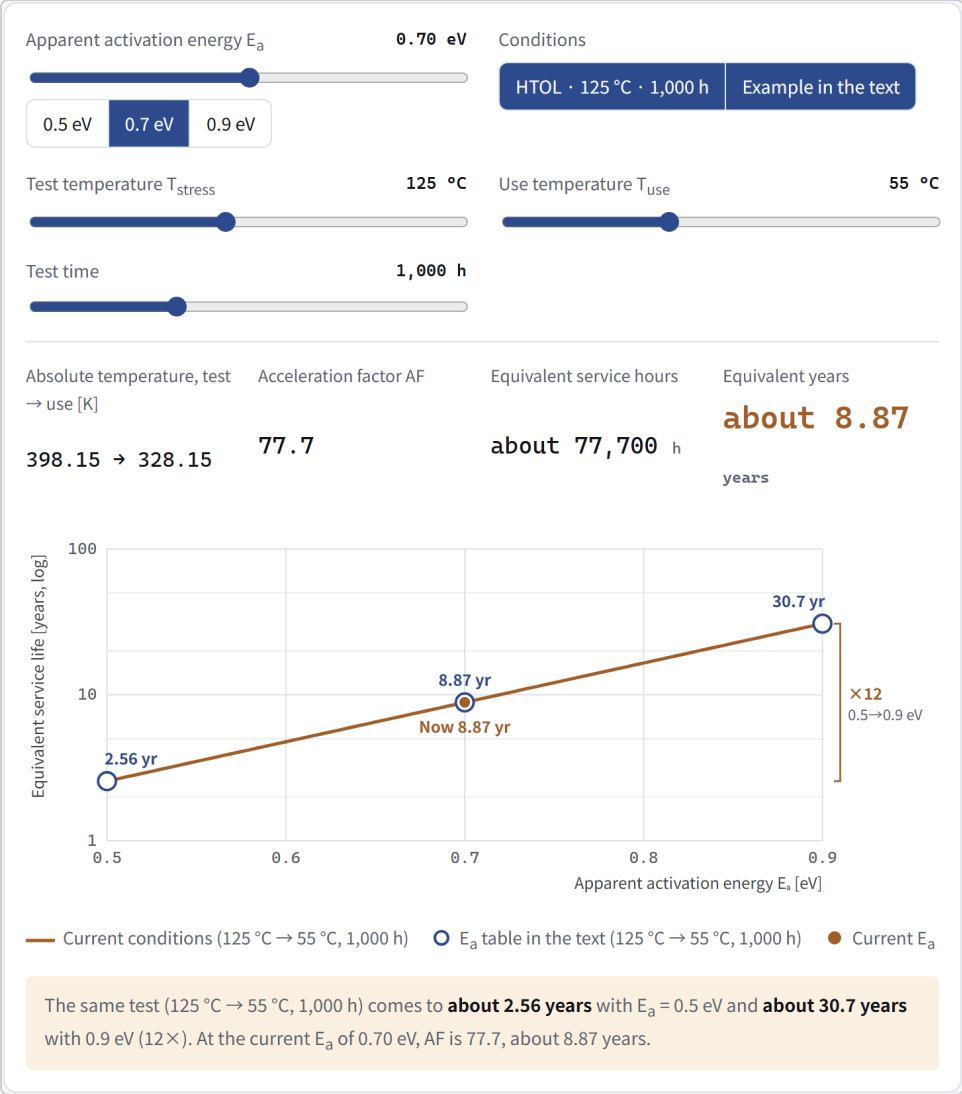


Figure 6-3. At an activation energy of 0.7 eV, a test at 125 °C and use at 55 °C, the acceleration factor is about 77.7, and 1,000 hours of HTOL equals about 8.87 years. At 0.5 eV it is about 2.56 years; at 0.9 eV, about 30.7 years. — In the HTML edition you can change the values here (HTML edition, Session 6, Section 7).

---

8. Heterogeneous integration and the chiplet standard (UCIe)

The dream of a chiplet ecosystem

If chiplets could be connected through a standard interface, **systems could be built by combining chiplets made by different companies.** You would assemble a chip the way you assemble a computer.

UCIe (Universal Chiplet Interconnect Express) is the standard for this. It places PCIe/CXL protocols on top of a die-to-die physical layer and specifies separate versions by package type (standard package / advanced package). An industry consortium announced its launch in March 2022, and version 2.0 in August 2024 added 3D packaging aimed at hybrid bonding (UCIe-3D) and a management and test architecture that spans chiplets. As of October 2026, the latest edition on the official specifications page is 3.0, released on August 5, 2025. 3.0

added data rates of 48 GT/s and 64 GT/s, doubling the top speed of 32 GT/s in 2.0.

Understanding it through a software analogy

CHIPLET CONCEPT	SOFTWARE COUNTERPART
Chiplet	Microservice
UCIe	Standard API / RPC protocol
Interposer	Service mesh / network
Heterogeneous integration	Polyglot architecture (the best language for each service)
Die-to-die latency	Network hop latency

And the trade-offs are the same. A monolith gets internal communication for free but has to be deployed whole; splitting it allows reuse and independent development but costs something every time a boundary is crossed.

Why it is still hard

1. **Testing and quality assurance.** If one chiplet is defective, the whole package already assembled is lost. Complete KGD assurance before assembly is essential, but it is hard to test fine, high-speed interfaces completely before assembly.
2. **Liability.** If you combine chiplets from several companies and the system does not work, whose fault is it? A real barrier to a commercial ecosystem.
3. **Integrated thermal and power design.** Each chiplet produces a different amount of heat, and power delivery has to be designed at the package level.
4. **Physical compatibility.** Even with the same protocol, the bump pitch, die thickness and package process have to match. **A standard that fixes only the protocol is not enough.**
5. **EDA tools.** Tools for designing and verifying several dies together (multi-die STA, coupled thermal-electrical simulation) are still maturing.

The reality: as of 2026, chiplets are used mainly **within one company** (in-house chiplets). A truly open chiplet market has not formed yet. But as standardization proceeds, the direction is clear. Version 2.0's addition of a test and management architecture to the standard is aimed at problem 1 above (testing and quality assurance).

---

9. Signal integrity — familiar ground if you have a communications background

As wires multiply into the tens of thousands and speeds rise, an electrical signal is no longer "0 and 1" but **a high-frequency transmission-line problem.**

PHENOMENON	CAUSE	REMEDY
Insertion loss	Conductor skin effect, dielectric loss. Worse at higher frequency	Low-loss materials, equalization (EQ)
Reflection	Impedance discontinuities	Impedance matching, termination resistors
Crosstalk	Capacitive and inductive coupling between neighboring wires	Wire spacing, ground shielding, optimized wire ordering
Simultaneous switching noise (SSN)	Many outputs switch at once and the power network shakes	Decoupling capacitors, power network design
Jitter	Fluctuation in clock and data timing	CDR, equalizers
Skew	Arrival times misaligned by differences in wire length	Length matching

The evaluation tool: an eye diagram is drawn by cutting the received waveform into one-bit lengths and overlaying them. The larger the open "eye" in the middle, the larger the margin before 0 and 1 are misread; as loss, reflection, noise and jitter grow, the eye closes. The remedies use the ideas of digital communication as they are. The CDR (clock and data recovery) in the table above is a circuit that recovers the clock timing from the transition points of the received data waveform without a separate clock line. Among equalizers, a CTLE (continuous-time linear equalizer) boosts again, with an analog filter at the receiver, the high-frequency components the channel ate away, and a DFE (decision feedback equalizer) estimates and subtracts the tails (intersymbol interference) left by preceding bits already decided as 0 or 1. Pre-emphasis, which boosts the signal at the moment of transition before sending it from the transmitter, and coding, which plants rules in the signal, serve the same purpose.

> **This completes the logic of Session 3.** Why did HBM choose width over frequency? Because every problem in the table above worsens with frequency. Widening the bus gives the same bandwidth at a lower frequency, easing signal integrity and power problems together. **HBM is a packaging answer to the signal integrity problem.**

---

10. Common misconceptions

Misconception 1: "The back end is easier than the front end" → Hybrid bonding demands angstrom-level flatness and nanometer alignment. Advanced packaging demands front-end-level precision.

Misconception 2: "2.5D is a kind of 3D, so they are roughly the same" → Their thermal characteristics are completely different. That difference decides the range of feasible designs.

Misconception 3: "Splitting into chiplets always makes it cheaper" → There are the area and power overhead of interface circuits, assembly costs and assembly yield loss. If the die is small, splitting is a loss.

Misconception 4: "Memory companies make all of HBM" → The memory companies make the HBM stacks, but the 2.5D packaging that joins them to the GPU is done by foundries (mainly TSMC). **The supply chain is split, and both sides can be bottlenecks.**

Misconception 5: "Now that UCIe is out, the era of buying chiplets and assembling them is near" → A protocol standard alone is not enough. Testing, quality assurance, liability and physical compatibility remain.

Misconception 6: "A TSV is just a hole filled with copper" → Stress from thermal expansion mismatch shifts the characteristics of the surrounding transistors and creates the design constraint called the keep-out zone. It is a mechanical problem, not an electrical one, that constrains the design.

Misconception 7: "Packaging capacity is easier to add than fab capacity" → Advanced packaging equipment also has long lead times and needs a yield ramp. In practice it has been a bottleneck of AI chip supply.

Misconception 8: "Today's HBM is stacked with hybrid bonding" → As of October 2026, volume-production HBM and the customer samples manufacturers have announced are stacked with TSVs and microbumps, with the gaps between chips filled by MR-MUF or TC-NCF. Hybrid bonding is a next-generation candidate for higher layer counts (section 6).

Misconception 9: "It passed 1,000 hours of HTOL, so its lifetime is nine years" → That conversion hangs on a single activation energy E_a, and E_a differs by failure mechanism. Under the same conditions, 0.5 eV gives about 2.6 years and 0.9 eV about 31 years (section 7, reliability).

---

11. Quantitative exercises

Exercise 1 — Chiplet gains and losses, the full version With $D_0 = 0.1/\text{cm}^2$, split a total area of 800 mm^2 into (a) 1, (b) 2, (c) 4 and (d) 8 pieces, and for each find the following. Write costs as multiples of W, the price of one wafer (the same symbol as in Session 4; not the power unit W, the watt).

- Yield per die and good dies per wafer
- Interface overhead (+5% area per additional split)
- Assembly yield (×0.99 for each additional die)

- Total cost

Calculate these and find the optimal number of splits. How does the optimum shift if D_0 is changed to 0.05 or 0.2?

Exercise 2 — Wiring energy Assume the parasitic capacitance of a 10mm wire (substrate) and a 50μm wire (TSV) is proportional to length, and use $E = \frac{1}{2}CV^2$ to find the ratio of energy per bit. What is the power difference when transmitting 10^{12} bits per second?

Exercise 3 — Interposer area With a GPU die of 800 mm^2 and eight HBM stacks (about 100 mm^2 each), estimate the interposer area needed, allowing for spacing between the stacks. How many times the reticle limit is it? Which CoWoS variant is needed?

Exercise 4 — The 3D heat problem If one chip has a power density of 1 W/mm^2, what is the heat flux per unit footprint when chips are stacked two and four high? Allowing for the thermal resistance of passing through 50μm of silicon, roughly estimate the temperature difference between layers. (Thermal conductivity of silicon about 150 W/m·K)

Exercise 5 — Counting the wires How many data wires alone are needed to attach eight HBM4 stacks? If power, ground and control are assumed to be twice the data wires, how many in total? If the HBM stacks are attached four each to two opposite sides of the GPU die (each side about 30 mm), and these wires come out in a single row along the edges of those two sides, what pitch is needed?

---

12. Discussion questions

1. If the field reduction of High-NA EUV raises the incentive to choose chiplets, are scaling and integration rival strategies or complementary ones?
2. How does the trend of foundries absorbing packaging change the OSAT industry? What is left for OSATs?
3. What has to be solved first for an open chiplet market to form? Is there anything to learn from the history of software ecosystems?
4. Is the heat problem of 3D logic stacking one that cooling can solve, or one that has to be avoided by design?
5. Can glass substrates replace silicon interposers? If they do, what changes will follow in the industry's structure?
6. Seeing that the bottleneck of AI chip supply has moved from logic → CoWoS → HBM, where do you think the next bottleneck will be?

---

13. Further study

- **Standard textbook:** Lau, *Semiconductor Advanced Packaging* (Springer, 2021) and the same author's series of books on 3D ICs and fan-out — the most systematic body of material in this field
- **Introductory series:** SK hynix Newsroom, "Semiconductor Back-End Process," 11 episodes (2022-09 to 2023-09 in the Korean original) — from Episode 1: Understanding Semiconductor Testing to Episode 11: Reliability Tests and Standards for Semiconductor Packages. It covers, in order, testing, package types, conventional and wafer-level processes, materials and reliability. Check its figures and terms against this book's primary sources as you read
- **Hybrid bonding:** public presentations from imec and CEA-Leti, papers from ECTC (Electronic Components and Technology Conference)
- **Industry technical material:** TSMC's 3DFabric (CoWoS/InFO/SoIC) technology pages, Intel's advanced packaging (Foveros/EMIB) pages, Siemens EDA's 3D IC guide to heterogeneous integration
- **Chiplet standard:** UCIe specifications (1.0–3.0, available on request)
- **Reliability and heat:** JEDEC JESD47 (the backbone of qualification testing; the text is based on JESD47I of July 2012, and the current edition is JESD47N), JEDEC JESD51-12 (guidelines for reporting and using thermal information), Renesas TB379, "Thermal Characterization of Packaged Semiconductor Devices" — explains θ and ψ with drawings of the test boards
- **Signal integrity:** Bogatin, *Signal and Power Integrity — Simplified* — well regarded for its intuitive explanations. Hall & Heck, *Advanced Signal Integrity for High-Speed Digital Designs* (advanced)
- **Conferences:** ECTC, IEEE 3D-IC Conference, Hot Chips (product-level architecture disclosures)

Self-study appendix A — solutions to the quantitative exercises

Solution to Exercise 1 — Chiplet gains and losses, the full version

▾ Solution

Setup: a total silicon area of 800 mm² is split into n pieces. Each split adds 5% interface area overhead cumulatively (total area $= 800 \times [1 + 0.05(n-1)]$), assembly yield is 0.99^{n-1}, and $D_0 = 0.1/\text{cm}^2$. The cost is the silicon cost of one good system, written as a multiple of W, the price of one wafer ($c_{\text{assembly}} = 0$ in the exact comparison formula of section 2). The W in the tables below is not the watt.

$$\text{GDPW}(A) = \frac{\pi d^2 / 4}{A} - \frac{\pi d}{\sqrt{2A}} \approx \frac{706.9}{A} - \frac{94.25}{\sqrt{2A}} \quad (d = 30 \text{ cm},\ A \text{ in cm}^2)$$

The values in the tables were computed with unrounded π. Computing with the rounded coefficients 706.9 and 94.25 changes the last digit in a few cells (for example, GDPW 466.3 for $n = 8$, and a cost of $0.03434\,W$ for $n = 1$ at $D_0 = 0.1$). The conclusions do not change.

At $D_0 = 0.1$

n	CHIPLET AREA	GDPW	YIELD	GOOD DIES	SYSTEMS/WAFER	ASSEMBLY YIELD	EFFECTIVE	COST
1	8.00 cm²	64.8	0.449	29.1	29.1	1.000	29.1	$0.03435\,W$
2	4.20	135.8	0.657	89.2	44.6	0.990	44.2	$0.02264\,W$
4	**2.30**	263.4	0.795	209.3	52.3	0.970	50.8	$0.01970\,W$
8	1.35	466.2	0.874	407.4	50.9	0.932	47.5	$0.02107\,W$

The optimum is $n = 4$, a 42.6% saving over a single die. What matters is that it gets worse again at $n = 8$ — interface overhead and assembly yield loss begin to eat up the yield gain.

Change D_0 and the optimum moves — the key finding of this exercise

n	$D_0 = 0.05$	$D_0 = 0.1$	$D_0 = 0.2$
1	$0.02302\,W$	$0.03435\,W$	$0.07644\,W$
2	0.01836	0.02264	0.03446
4	0.01756	0.01970	0.02479
8	0.01969	0.02107	0.02412
Optimum	$n = 4$	$n = 4$	$n = 8$
Saving	23.7%	42.6%	68.5%

This table compares only the four values the exercise picked. Widening the calculation to every integer number of splits from 1 to 16, the optimum is $n = 3$ at $D_0 = 0.05$ (0.01754 W, a 23.8% saving), $n = 4$ at $D_0 = 0.1$ (unchanged; almost the same as 0.01971 W for $n = 5$), and $n = 6$ at $D_0 = 0.2$ (0.02364 W, a 69.1% saving). The direction is the same: the more defects, the finer the split. Near the optimum, though, the curve is flat. At $D_0 = 0.1$, $n = 4$~6 all come within 2% of the optimum.

Three conclusions

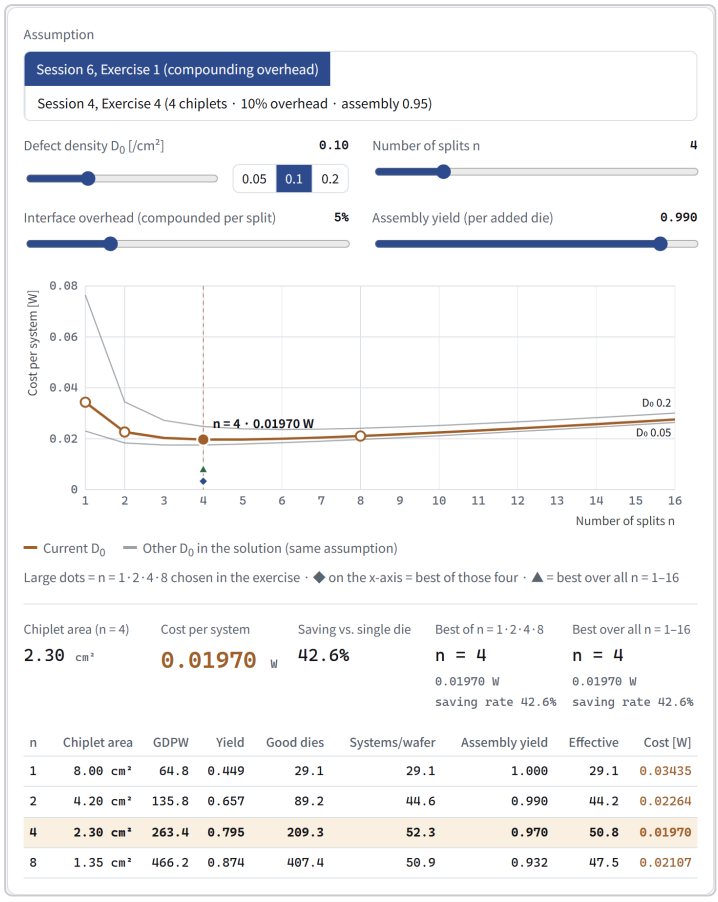
1. **The more defects there are, the more it pays to split finely.** At $D_0 = 0.2$ the optimal split moves to 8, and the saving reaches 68.5%.
2. **With few defects, the advantage of chiplets fades.** At $D_0 = 0.05$ the saving shrinks to 23.7%, and adding assembly and interposer costs can reverse it.
3. **An optimal number of splits exists.** Split without limit and the overhead wins. The shape of this curve is why real products cluster in the range of 2–8 chiplets.

Implications — reading it as industrial strategy

In cost terms, chiplets are **a tool that lets you use the latest node early, while its yield is still poor.** That those trying to enter the latest node first, and AI accelerators that need large dies, adopted chiplets early fits this direction. Conversely, for small chips at mature nodes the yield gain is small, and on cost alone chiplets are likely to lose. But what this exercise shows is a single cost motive coming from yield. Area beyond the reticle limit (pressure ②), heterogeneous integration that uses a different process for each block (pressure ③) and design reuse are also reasons to use chiplets (the same scope as the solution to Exercise 4 of Session 4), so this calculation alone cannot decide where chiplets are used and where they are not.

> Combine this with Exercise 5 of Session 5 and the picture is complete. High-NA shrinks the largest die that can be printed in one shot to 429 mm². If you need a circuit of 800 mm², you must choose between stitching and partitioning, and this exercise shows that under its assumptions **partitioning is also the economical choice.** The incentive to choose chiplets grows accordingly.

The figure below shows the assumptions of this exercise with $D_0 = 0.1$ and $n = 4$. In the screen edition you can change D_0, the number of splits, the overhead and the assembly yield, and you can also switch to the assumptions of Exercise 4 of Session 4 (10% overhead per chiplet, an assembly yield of 0.95 applied once).



n	Chiplet area	GDPW	Yield	Good dies	Systems/wafer	Assembly yield	Effective	Cost [W]
1	8.00 cm²	64.8	0.449	29.1	29.1	1.000	29.1	0.03435
2	4.20 cm²	135.8	0.657	89.2	44.6	0.990	44.2	0.02264
4	**2.30 cm²**	**263.4**	**0.795**	**209.3**	**52.3**	**0.970**	**50.8**	**0.01970**
8	1.35 cm²	466.2	0.874	407.4	50.9	0.932	47.5	0.02107

Figure 6-4. System cost (as a multiple of the wafer price) when 800 mm² of silicon is split into n dies. With $D_0 = 0.1$, 5% overhead per split and assembly yield 0.99 for each added die, n = 4 is cheapest at 0.01970, 42.6% cheaper than a single die. — In the HTML edition you can change the values here (HTML edition, Session 6, Solution to Exercise 1).

Solution to Exercise 2 — Wiring energy

▾ Solution

If parasitic capacitance is proportional to length,

$$\frac{C_{\text{substrate}}}{C_{\text{TSV}}} = \frac{10\ \text{mm}}{50\ \mu\text{m}} = 200\ \text{times}$$

From $E = \frac{1}{2}CV^2$, at the same voltage **the energy per bit is also 200 times.**

At 10^{12} bits per second

Taking the energy per bit on the TSV side as 0.05 pJ/bit:

PATH	PJ/BIT	POWER AT 1 TBIT/S
TSV (50 µm)	0.05	**0.05 W**
Substrate wiring (10 mm)	10	**10 W**

A difference of about 10 W. On its own this does not look large. Scale it up to real size,

though, and the story changes.

When one HBM stack runs at 4 TB/s $= 3.2 \times 10^{13}$ **bit/s**

PATH	POWER
Interposer/TSV level	1.6 W
Substrate wiring level	**320 W**

With eight stacks it is **12.8 W vs 2,560 W.** As seen in section 7, one such package draws power in the kilowatt range (the worked example in section 7 also used 1,000 W). Spending 2.56 kW on memory wiring alone would exceed the whole power budget of the package, so under this assumption the design does not work.

Conclusion: putting HBM on the interposer right next to the GPU is not a convenience but **a necessity.** If you cannot shorten the distance, that bandwidth cannot be delivered within the power budget.

This is where it connects to Session 3. Exercise 2 of Session 3 said the choice of "many slow pins" was made because of power. But using many pins needs room to attach them, and making that room needs an interposer. **Bandwidth → power → distance → packaging** line up in one chain.

> **Caution:** the factor of 200 comes from assuming the same capacitance per unit length and the same voltage. Real wires differ in cross-section, insulator, parasitics and driver circuits, so this ratio does not come out as is. To note just one direction of difference: substrate I/O often uses a higher voltage swing for signal integrity, so the gap can widen further through the V^2 term.

The figure below is for 10 mm of substrate wiring vs 50 µm of TSV, 0.05 pJ/bit on the TSV side and 10^{12} bits per second. In the screen edition you can change the two lengths, the energy per bit and the bit rate (including one HBM stack at 4 TB/s).



Figure 6-5. A 10 mm wire and a 50 µm TSV differ in length by 200 times, so energy per bit also differs by 200 times. At 0.05 pJ/bit for the TSV, 10^{12} bits per second costs 0.05 W versus 10 W. — In the HTML edition you can change the values here (HTML edition, Session 6, Solution to Exercise 2).

Solution to Exercise 3 — Interposer area

▾ Solution

Estimating the area needed

COMPONENT	AREA
GPU die	800 mm²
8 HBM stacks × 100 mm²	800 mm²
Subtotal	1,600 mm²
Allowance for spacing between stacks, molding and peripheral wiring (25%)	400 mm²
Total	**about 2,000 mm²**

Compared with the reticle limit

$$\frac{2{,}000}{858} \approx 2.3 \text{ times}$$

It cannot be made with a single exposure. The interposer itself is silicon, and the

wiring of a silicon interposer is also made by lithography, so it faces the same 858 mm^2 constraint.

Three solutions

APPROACH	WHAT IT DOES	CHARACTER
Stitching	Joins several fields to make a large interposer	Alignment accuracy and defects are the issue
Local bridges (CoWoS-L / EMIB family)	Instead of a full silicon interposer, small silicon bridges are inserted **only where connections are needed.** The layer around the bridge differs by structure. CoWoS-L puts the bridge (LSI) inside an RDL interposer, and that interposer is in turn mounted on the package substrate. Intel EMIB has no interposer. Intel describes it as "Silicon bridge embedded in package substrate for shoreline-to-shoreline connection"; the bridge joins the facing edges of two dies	Gets around the area constraint. The current mainstream direction
Large-area RDL substrate	Covers the whole with organic redistribution layers	Wiring density lower than silicon

Enlarging a silicon interposer as a whole adds a yield burden on top of the reticle limit. Let us only gauge how large it could get. Applying a logic die's $D_0 = 0.1/cm^2$ as is to a 2,000 mm^2 (20 cm^2) interposer gives $e^{-2.0} = 13.5\%$ by the formula of Session 4. There is no basis for assuming that a passive interposer carrying only wiring has the same real defect density as a logic die, so 13.5% is not an actual manufacturing yield but a sensitivity example under the assumption of the same defect density. The direction is **from "silicon everywhere" to "silicon only where needed."** TSMC keeping the silicon interposer (CoWoS-S) to 3.3 times the reticle and offering, for larger packages, the RDL-based CoWoS-R and CoWoS-L with local bridges (3.5 times, in volume production in 2024) follows this direction as well.

This is why AI accelerator supply runs into CoWoS capacity before logic wafers — the substance of the bottleneck foreshadowed in section 5 of Session 1.

Solution to Exercise 4 — The 3D heat problem

▾ Solution

Heat flux per unit footprint

Stack chips with a power density of 1 W/mm² and it multiplies directly.

STACK	HEAT FLUX PER FOOTPRINT
1 layer	1 W/mm²
2 layers	2 W/mm²
4 layers	**4 W/mm²**

For reference, 4 W/mm² = 400 W/cm². For a sense of scale, take an electric stove that puts out 2 kW from a burner 20 cm in diameter and divide the power by the burner area: about 6.4 W/cm² (a calculation with assumed values). The heat flux per footprint of a four-layer stack is more than 60 times that.

Thermal resistance of 50 µm of silicon (per 1 mm²)

$$R = \frac{t}{kA} = \frac{50 \times 10^{-6}}{150 \times 10^{-6}} = 0.333 \text{ K/W}$$

Temperature differences between layers in a four-layer stack (heat removed upward only, each layer dissipating 1 W/mm²)

The heat of the lower layers has to pass through all the layers above.

SEGMENT	HEAT PASSING	ΔT
Layer 1 → layer 2	1 W	0.33 K
Layer 2 → layer 3	2 W	0.67 K
Layer 3 → layer 4	3 W	1.00 K
Layer 4 → heat-removal surface	4 W	1.33 K
Total		**3.33 K**

An unexpected result: the silicon alone accounts for only 3.3 K. Silicon conducts heat very well.

Then where does the heat problem of 3D stacking come from? — the interfaces

Silicon is not all that lies between layers. There are microbumps, underfill and adhesive layers, and their effective thermal conductivity is about 1–5 W/(m·K), **around 1/50 of silicon's.** Taking a thickness of 20 µm and $k = 3$ W/(m·K),

$$R_{\text{interface}} = \frac{20 \times 10^{-6}}{3 \times 10^{-6}} = 6.67 \text{ K/W}$$

How many interfaces are there? With four layers there are three between layers, and one more bonding layer between the top die and the heat-removal surface. Here all four are taken to be the same material (20 µm, $k = 3$). The heat crossing each interface is 1, 2, 3 and 4 W, as for the silicon segments, so $(1 + 2 + 3 + 4) \times 6.67 = 66.7$ K. If the top bond is counted separately as a different material such as a TIM and only the three internal interfaces are included, $(1 + 2 + 3) \times 6.67 = 40$ K, and 43.3 K with the silicon added.

Silicon 3.3 K vs interfaces 66.7 K — a factor of 20. Together, the temperature difference between the bottom layer and the heat-removal surface is about 70 K (four-interface model).

Two conclusions

1. **The thermal bottleneck of 3D stacking is not the silicon but the bonding interfaces.** So the remedy lies less in thinning the silicon than in lowering the thermal resistance of the interfaces.
2. Hybrid bonding is a strong candidate. Instead of layers of bumps and underfill, it joins copper and dielectric directly, so the bonding layer becomes much thinner. Hybrid bonding is usually introduced as "a technology for shrinking pitch," but **it also matters a great deal from a thermal point of view.** That is as far as this calculation goes, however. Removing the interfaces in the figure below drops 70 K to 3.3 K, but that is an idealization that sets the thermal resistance of all four interfaces, up to the top heat-removal bond, to 0. A hybrid bond still leaves two bonded surfaces, copper and dielectric (section 6), and their effective thermal conductivity and the copper area fraction were not calculated here. Volume-production HBM is also lowering thermal resistance by improving gap-fill materials (section 6; about 17% for SK hynix's 12-high HBM4E samples). To state the actual improvement, you would have to recalculate with the material values of the bonding layer.

> This logic of a thermal-resistance chain extends directly into Session 7. The path from chip to coolant has the same structure, and there too the bottleneck is the layer that conducts heat worst (the TIM).

The figure below shows the temperature difference that builds up in each segment for four layers of 1 W/mm² chips, 50 µm of silicon and interfaces of 20 µm at 3 W/(m·K). In the screen edition you can change the number of layers, the power density, the thicknesses and thermal conductivities, and remove the interfaces.

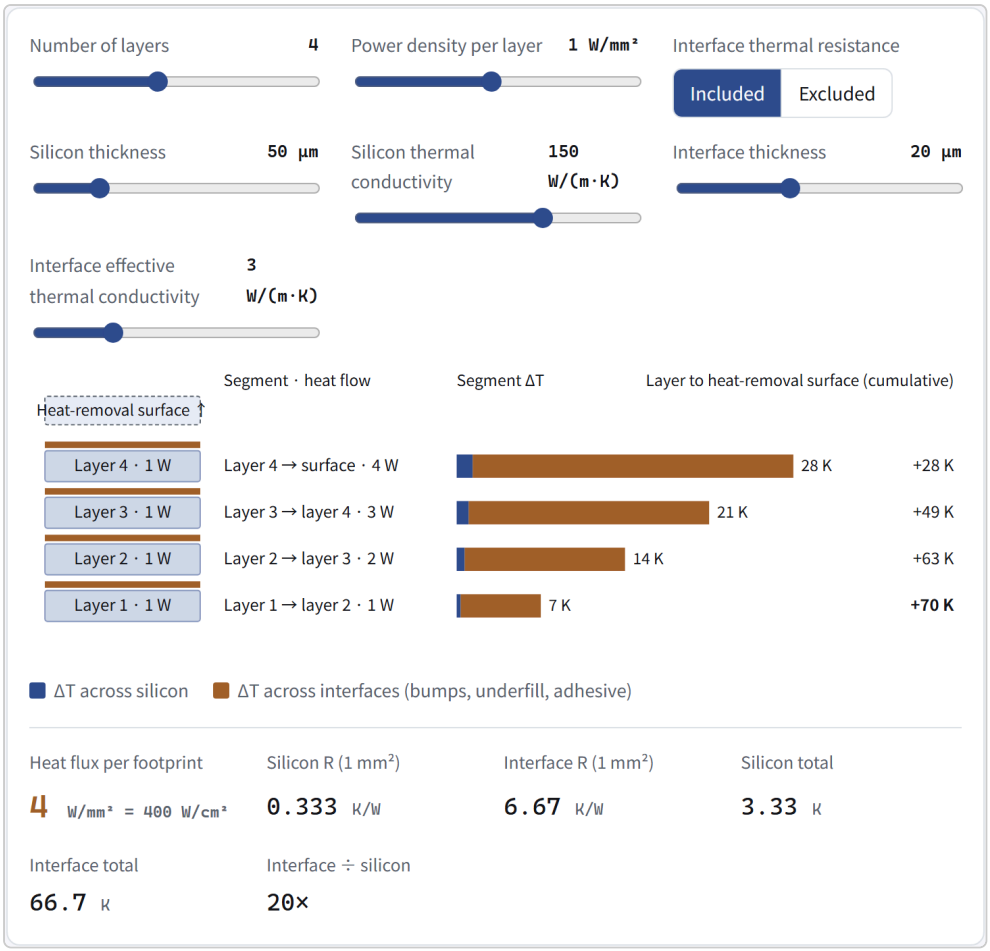


Figure 6-6. Stacking 4 layers of 1 W/mm² chips produces a total temperature difference of 3.33 K across the silicon (50 µm) but 66.7 K across interfaces such as bumps and underfill (20 µm, 3 W/(m·K)), 20 times as much. — In the HTML edition you can change the values here (HTML edition, Session 6, Solution to Exercise 4).

Solution to Exercise 5 — Counting the wires

▾ Solution

Data wires

Taking the data bus of one HBM4 stack as 2,048 bits,

$$8 \text{ stacks} \times 2{,}048 = 16{,}384 \text{ wires}$$

Assuming power, ground and control are twice the data wires

$$16{,}384 \times 2 = 32{,}768 \text{ wires}$$

$$\text{Total} = 16{,}384 + 32{,}768 = 49{,}152 \text{ wires}$$

What if they come out in a single row along the edges of two sides (30 mm each)?

With four stacks on each of two opposite sides, as the exercise sets out, each side carries 49,152 ÷ 2 = 24,576 wires.

$$\text{Pitch} = \frac{30\ \text{mm}}{24{,}576} = 1.22\ \mu\text{m}$$

(If all eight stacks were attached to a single side, 49,152 wires would have to fit into 30 mm, giving 0.61 µm.)

That is unrealistic with today's connection methods. The pitch of the microbumps that join the GPU and HBM to the interposer is about 40 µm (section 6), and 1.22 µm is a 33rd of that. On the pitch number alone it falls within the sub-10 µm range claimed for hybrid bonding, but as seen in section 6, hybrid bonding is not the volume-production method for HBM connections as of October 2026, and that pitch is the pad pitch of an area array, not a value that assumes a single row along an edge.

So what is actually done — a two-dimensional area array

The connections are laid out not in one row along the edge but **as a grid over the whole underside of the die.** The area needed is

PITCH	AREA NEEDED	RELATIVE TO THE GPU DIE (800 MM²)
40 µm (microbumps)	$49{,}152 \times (0.04)^2 = 78.6\ \text{mm}^2$	9.8%
9 µm (hybrid bonding)	4.0 mm²	0.5%

Two conclusions

1. **The shift from peripheral I/O to area-array I/O was not a choice.** This number of connections cannot be pulled out in one row at microbump pitch. This is the fundamental reason flip chip and microbumps became the standard.
2. **Bump pitch presses down on the ceiling of bandwidth.** Under this exercise's assumption (connections three times the data wires), a 40 µm-pitch area array uses 9.8% of the die area. If the number of connections grows several-fold (a new HBM generation, more stacks), this share grows by the same factor and eats into the die. Hybrid bonding, which shrinks the pitch, is a strong path for that moment. But the real limit is set not by pitch alone but together by the interposer's wiring density, the HBM interface specification and the yield of the bonding process.

> Here is the price of choosing "wide and slow" in Exercise 2 of Session 3. Wide means many wires, many wires means pitch, and pitch means bonding technology. **The memory bandwidth roadmap and the packaging roadmap are the same curve.**

Self-study appendix B — leads for the discussion questions

QUESTION TYPE	CORE AXIS	LEADS
Why the back end suddenly became important	**Diminishing returns to scaling**	While the gains from the front end shrank and its costs soared, the gains available from packaging (bandwidth, integration) are still in an early phase. The marginal returns crossed over; packaging did not suddenly get better
Will the chiplet standard (UCIe) succeed?	**Who benefits**	Standardization **favors late entrants and small design firms** and disadvantages the leading integrators. Leaders have an incentive to keep their own interfaces, so the standard is likely to settle not as "what everyone uses" but as "the alliance of those who are not the leader." Compare it with the history of PC and server standards
OSAT vs foundry	**Is packaging a process or an assembly step?**	The moment advanced packaging uses lithography and etching, it is no longer assembly but **an extension of the front end.** That is why foundries are absorbing this area and OSATs are being pushed toward mature packages. It is a question of where value added moves
Is heat the ceiling of 3D stacking?	**Which level's thermal resistance**	Exercise 4 gives the answer — silicon is not the problem; **the interfaces** are. Lowering interface thermal resistance with hybrid bonding and the like raises the ceiling. How far it rises has to be calculated with the material values of actual bonding layers. Stacking logic on logic (both high-power) is still hard, though, and the realistic answer is an asymmetric design that **puts the lower-power part on top**
When will the supply bottleneck ease?	**Equipment and the learning curve**	CoWoS capacity can be expanded with capital, but hybrid bonding equipment has few suppliers and needs yield learning. The logic of Exercise 1 of Session 3 applies here too — **you can forecast only if you separate a capacity problem from a learning problem**

Self-study appendix C — key terms

TERM	ONE-LINE DEFINITION
Back end (assembly & test), packaging	The stage that cuts the wafer, turns chips into packages and tests them. Different from BEOL, the forming of wiring layers within wafer processing (Session 5)
Dicing	The step that cuts a wafer into individual dies
Flip chip	Flipping the die and attaching it directly to the substrate with bumps
Microbump	A fine solder bump connecting dies. Pitch about 40 µm
Hybrid bonding	Direct copper-to-copper bonding without bumps. Sub-10 µm pitch (TSMC SoIC). Used in CIS and 3D logic; for HBM, a next-generation candidate as of October 2026
MR-MUF / TC-NCF (mass reflow molded underfill / thermal compression non-conductive film)	The stacking methods of volume-production HBM. One stacks everything, reflows it all at once and fills the gaps with a liquid protective material; the other lays a film on each layer and thermocompresses it

TERM	ONE-LINE DEFINITION
Fan-in WLCSP / fan-out	Fan-in if the balls are only within the chip's area; fan-out if the wiring is spread beyond the chip over a reconstituted wafer of rearranged good dies (e.g., TSMC InFO)
Via-middle TSV	A TSV drilled after the transistors are made but before the wiring layers are built. Before that is via-first; after, via-last
WSS (wafer support system)	Handling a wafer to be thinned by attaching it to a carrier with a temporary adhesive, and removing it (debonding) after backside processing
KGSD (known good stacked die)	HBM tested as a whole stack. Memory makers ship it in this form
CoW-first / CoS-first (chip-on-wafer first / chip-on-substrate first)	2.5D assembly orders. Whether the chips are attached first onto the interposer wafer, or the interposer is first cut and attached to the substrate
W2W / D2W (wafer-to-wafer / die-to-wafer)	Joining wafers to wafers cannot pick good dies, so yields multiply as y^n; picking and joining only good dies leaves only the bonding yield
θJA · θJC · ψJT (junction-to-ambient / junction-to-case thermal resistance, junction-to-top characterization parameter)	The difference between junction temperature and a reference temperature, divided by power. θJA is a comparison value under standard test conditions; ψ is a characterization parameter, not a true thermal resistance
HTOL (high temperature operating life)	A qualification test that operates parts for 1,000 hours at a junction temperature of 125 °C or higher (JESD47I)
Acceleration factor	How many hours under use conditions one hour under test conditions corresponds to. In the Arrhenius equation it depends heavily on E_a
Interposer	An intermediate substrate on which several dies sit and are connected to each other
TSV (through-silicon via)	Wiring that passes vertically through silicon
TGV (through-glass via)	Wiring that passes through a glass substrate. A candidate to replace silicon
2.5D	Placing dies side by side on an interposer
3D stacking	Stacking dies vertically
CoWoS (chip-on-wafer-on-substrate)	TSMC's family of 2.5D packaging
EMIB / bridge (embedded bridge)	Connecting only the necessary sections with a small piece of silicon
RDL (redistribution layer)	A wiring layer that relocates pad positions
Heterogeneous integration	Integrating dies of different processes and functions in one package
UCIe (universal chiplet interconnect express)	The interface standard between chiplets
KGD (known good die)	A good die verified before assembly
Underfill	Resin that fills the space between bumps for mechanical reinforcement
TIM (thermal interface material)	The heat-transfer material between chip and heat sink
Signal integrity	The degree to which a signal is delivered without distortion or noise
Warpage	The bending of a package caused by differences in thermal expansion

Self-study appendix D — self-assessment

☐ I can distinguish 2D, 2.5D and 3D by **the way things are connected** → self-study guide ★

☐ I can show by calculation that the optimal number of chiplet splits shifts with D_0 → Exercise 1 ★

☐ I can give two reasons why splitting without limit is a loss → Exercise 1

☐ I can explain in terms of power **why putting HBM on an interposer is a necessity** → Exercise 2 ★

☐ I can state the problem of the interposer area exceeding the reticle limit and its three solutions → Exercise 3

☐ I can show by calculation that the thermal bottleneck of 3D stacking is **the interfaces, not the silicon** → Exercise 4 ★

☐ I can explain why hybrid bonding is a candidate **in both** pitch and heat, and where this book's calculation stops → Exercises 4 and 5

☐ I can explain from the wire count why the area array was unavoidable → Exercise 5

☐ I can state the physical meaning of the phrase "AI accelerator supply is held back by CoWoS" → Exercise 3

☐ I can calculate how far the stack yields of W2W and D2W diverge with the number of layers → section 6

☐ I can state the difference between θJA and ψJT, and how to find T_j with a series chain of thermal resistances → section 7

☐ I can show how much the equivalent use period of 1,000 hours of HTOL swings with E_a → section 7, reliability

---

Before the next session

This is where **the chip and the package** end. Session 7 is about what happens when tens of thousands of them are gathered together.

QUESTION THIS SESSION LEAVES	SESSION 7'S ANSWER
How does the thermal-resistance chain continue outside the package?	Chip → TIM → cold plate → coolant → cooling tower
How do you cool 120 kW in one rack?	Exercise 1 — why air cooling cannot do it
How does power get to the chip?	Exercises 2 and 4 — conversion losses and IR drop
What does all of this come up against in the end?	Power, land and the grid — **the ceilings of the physical world**

Sources

Introductory material (by topic)

TOPIC	LINK	HOW THIS BOOK USES IT
Back-end concepts	SK hynix Newsroom, "Semiconductor Back-End Process Episode 6: The Eight Steps of Assembling Conventional Packages" (the full series of 11 episodes: Episode 1 · Episode 2 · Episode 3 · Episode 4 · Episode 5 · Episode 7 · Episode 8 · Episode 9 · Episode 10 · Episode 11)	Basis for the back-end concepts of section 1. **An accessible series, recommended for preparation.** For factual statements the text relies as far as possible on the primary sources below. This series is the basis for the order of wafer test (Episode 1), package levels (Episode 2), and WSS, debonding, edge trimming and the fan-out reconstituted wafer (Episode 8). The Korean edition used the Korean versions of the series; this edition links to SK hynix's English versions of the same episodes
Packaging technology trends	ScienceON — Kim Heeju and Jung Jae Pil, "Artificial Intelligence Semiconductor and Packaging Technology Trend," *Journal of the Microelectronics and Packaging Society* 30(3), 2023 (in Korean)	An academic review of packaging technology trends in Korea
Packaging technology trends	ScienceON — Jeong Haksun and Kwon Jimin, "Advances and Research Trends in 3D Packaging-Based High-Bandwidth and Large-Capacity Memory Technologies for Next-Generation AI Hardware," same journal 33(2), 2026 (in Korean)	Recent trends in 3D packaging and memory technology
Dicing	ScienceON report — Lim Musaeng, "Laser Dicing Machines for Semiconductor Manufacturing Equipment" (our translation of the Korean title), ReSEAT analysis report, 2004 (in Korean)	A report on laser dicing equipment. Background to section 3
TSV, TGV, interposers	ScienceON — Seo Ilwoong et al., "Thermal Analysis of 3D package using TSV Interposer," same journal 21(2), 2014 (in Korean)	Thermal analysis of a 3D package with a TSV interposer. Background to sections 4 and 5 and to the heat discussion of section 7
TSV, TGV, interposers	ScienceON — Sun Sinhee et al., "Recent Research Trends in TGV and Cu-Filling for AI Semiconductor Packaging," same journal 31(4), 2024 (in Korean)	Literature on TGVs and glass substrates
Bonding	ScienceON — Joo Seunghwan et al., "Trends in Ultra-High Density Hybrid Bonding Stack Equipment Technology for High Bandwidth Memory (HBM) Semiconductors," same journal 32(1), 2025 (in Korean)	Literature on hybrid bonding equipment for HBM. Background to section 6
2.5D/3D packaging and the HBM interposer	Ovaga — "Advanced Packaging & Heterogeneous Integration"	An English overview of advanced packaging and heterogeneous integration. Corresponds to sections 7 and 8. When checked on 2026-10-08 the page returned only a JavaScript gate, so the title could not be confirmed and no link is given

TOPIC	LINK	HOW THIS BOOK USES IT
2.5D/3D packaging and the HBM interposer	Granite River Labs — "Signal Integrity and Protocol Testing of DDR"	**Practical background to the signal integrity of section 9.** Material from a measurement standpoint
2.5D/3D packaging and the HBM interposer	Siemens EDA — "Comprehensive 3D IC guide to heterogeneous integration" (2025-05-19)	**An EDA vendor's 3D IC design guide. Direct background to section 8 on heterogeneous integration and the design-tool problem**

Verified (2026-08-02)

- The structural differences among the CoWoS variants (CoWoS-S / -R / -L) — compared against TSMC technical material and several industry analyses
- The principle of hybrid bonding (direct copper-to-copper bonding + dielectric bonding, diffusion bonding on anneal) — standard technical fact
- The 2.5D / 3D distinction and the point that an HBM system is a "3D inside 2.5D" structure — standard technical fact
- The TSV process flow and the keep-out zone caused by CTE mismatch — standard technical fact
- The structure in which NVIDIA has secured a large part of CoWoS capacity and advanced packaging capacity constrains AI accelerator supply — several industry analyses agree. **But specific capacity figures and shares vary widely between sources and are not stated as figures in the text**
- The lithography reticle field limit and the field reduction of High-NA — the same as Session 5 (confirmed with official ASML material on 2026-10-08, below)

Verified (2026-10-08)

- Reticle field — ASML TWINSCAN NXE:3600D product page (maximum exposure field of the NXE platform 26 mm × 33 mm), ASML, "5 things you should know about High NA EUV lithography" (the exposure field of EXE is half that of NXE; quoted wording rechecked 2026-10-22). 858 and 429 mm² are calculated values
- Package flow and functions — Intel, "How Silicon Die Become Chip Packages" (assembly and test; the package protects against heat and the outside and connects; TIM and lid attach; high-voltage, high-temperature testing). The four requirements of a package (interconnection, powering, cooling, protection) were confirmed from the description ("Electronic packaging is defined as interconnection, powering, cooling, and protecting semiconductor chips for reliable systems.") on the Springer book page for Tummala, Rymaszewski and Klopfenstein (eds.), *Microelectronics Packaging Handbook: Semiconductor Packaging*, 2nd ed. (Chapman & Hall, 1997; hardcover ISBN 978-0-412-08441-6). The edition (2nd), year (hardcover published 1997-01-31; the e-book appeared separately on 2013-11-27), editors and ISBN were confirmed from this publisher's catalog; the text of the book itself was not checked
- EMIB structure — Intel Foundry, "Advanced Packaging Innovations | Chip Packages" ("Silicon bridge embedded in package substrate for shoreline-to-shoreline connection," in

volume production since 2017; wording rechecked 2026-10-22)

- Qualification tests and Arrhenius — JEDEC JESD47I (2012-07) Table 1 (HTOL, ELFR, HTSL), Table 2 (PC, HTSL, THB, HAST, TC, UHAST, AC) and their notes, note a to Table 1 (acceleration factor 78.6 at 0.7 eV and 125→55 °C, about 9 years). The numbers in the text's table were checked against this edition. The acceleration factor was recalculated with $k = 8.617 \times 10^{-5}$ eV/K (77.66). According to the JEDEC JESD47 page, the current edition is JESD47N (2026-08). **The conditions of the latest edition were not checked**
- Definitions of thermal resistance — TI, "One-Page Thermal Reference Sheet," SLRR012 (θJA is defined by JEDEC 51-2A, is for comparison and not for calculating T_j, and T_j is estimated with ψJT), Renesas TB379 (test boards 1S0P and 1S2P, 1S2P lowering θJA by roughly 20–40% or more, ψ not a true thermal resistance). The original text of JEDEC JESD51-12 was not seen
- HBM stacking methods — SK hynix, "SK hynix Ships Samples of 12-Layer Next-Gen 'HBM4E'" (2026-06-18) (Advanced MR-MUF, 12-high 48 GB, thermal resistance improved by about 17% over HBM4; checked against the Korean version of the release), SK hynix, "The value of interconnection technology driving advanced packaging, and SK hynix's packaging innovation" (our translation of the Korean title; 2023-08-18, in Korean) (definitions of TC NCF and MR-MUF, volume production of CIS with hybrid bonding, the 2022 hybrid-bonding test of an 8-high HBM2E). Samsung's TC-NCF is in the sources of Session 3 (Samsung Electronics press release of 2024-02-27). Micron's method and the schedule for adopting hybrid bonding had no primary source and were not written
- KGSD — SK hynix Hot Chips 28 tutorial, "The Era of High Bandwidth Memory" (2016) (Known Good Stacked Die, molded KGSD)
- TSV order — specification of US 11,011,499 B2 (definitions of via first/middle/last; HBM and HMC as typical examples of via first and via middle). That memory TSVs are via-middle is confirmed by this source only as far as "via-first or via-middle"
- CoW-first order — abstract of Ho et al., "2.5D through silicon interposer package fabrication by chip-on-wafer (CoW) approach," EPTC 2014. The warpage of CoS-first is explained from the principle of the difference in thermal expansion coefficients; analyst material is not cited
- TSMC 3DFabric — 3DFabric, InFO (Integrated Fan-Out, shipped in high volume since 2016; InFO_MS as a cost/performance alternative to CoWoS), CoWoS (volume production in 2012; -S 3.3 times and ~2,700 mm^2; -R in volume production in 2023; -L 3.5 times in volume production in 2024; demand rising since generative AI and capacity expanding), SoIC (bond pitch from under 10 μm; 3 nm chip stacking in volume production in 2025; SoIC-X wafer-on-wafer suited to high-yield nodes and the same die size) — all checked 2026-10-08; quoted wording rechecked 2026-10-22
- UCIe — UCIe specifications page (1.0–3.0; 48 and 64 GT/s in 3.0; UCIe-3D and DFx in 2.0), list of UCIe press releases (launch 2022-03-02, 2.0 on 2024-08-06, 3.0 on 2025-08-05)
- Glass substrates — Intel glass substrate announcement (2023-09-18) (target of the second half of the decade; quoted wording rechecked 2026-10-22 against the same release on Intel's investor site), Intel and Lens Technology collaboration (2026-07-24) (a collaboration to accelerate development)

- WSS process order — checked against the original of episode 8 of the SK hynix Newsroom series "Semiconductor Back-End Process" (edge trimming before carrier bonding; carrier bonding → backgrinding → backside bumps → carrier debonding; four debonding methods; the Korean edition checked the Korean version). The paragraph in section 4 reorganizes this content in this book's own order and classification
- Calculations — the W2W and D2W yield table (including the case with the bottom wafer in the denominator), the thermal-resistance chain example, the acceleration-factor table, the unit conversion in the yield comparison of section 2, the GDPW of Exercise 1 (the exact-π formula and the rounded-coefficient formula), the number of interfaces in Exercise 4 (four and three) and the pitch in Exercise 5 (two sides and one side) were recalculated in Python. The inputs (y, y_{bond}, θ for each stage, P, T_a) are all assumed values

Unconfirmed or reserved (not asserted in the text)

- **The size of CoWoS capacity and the allocations by company** — reports vary widely and official confirmation is hard. Only the structural fact (that it is a bottleneck) and TSMC's official statement that it is expanding capacity are used
- **When glass substrates will be commercialized** — apart from Intel's target of the second half of the decade, no official announcement of a volume-production schedule was found as of October 2026
- **When hybrid bonding will be applied to volume-production HBM** — as of October 2026, no manufacturer announcement of its use in volume-production HBM was found (the same as Session 3)
- **The test conditions of the latest JESD47 edition (JESD47N, 2026-08)** — the table in the text is based on JESD47I of 2012 and was not checked against the latest edition
- **The fuse method of memory repair, volume-production burn-in conditions** — these are company-specific processes and no primary source was found, so they are not described specifically in the text
- **The actual adoption of UCIe** — the standard is published, but commercial combinations across different vendors are limited. Described as "the direction is clear but the market has not formed"
- **HBM stack die area, GPU die area** — the figures in the exercises of section 11 are **assumed values for calculation practice**, not the actual dimensions of any particular product
- **Thermal conductivity of silicon, 150 W/m·K** — an approximation that varies with temperature and doping

SESSION 7

AI Data Centers

KEY TAKEAWAY

The heat and power problems that began at the chip become the final ceiling at the level of the building. As the power density of AI racks jumped to ten times that of a conventional data center, cooling, power distribution and the building itself all had to change, and the constraint on AI infrastructure is turning from chip supply into **grid connection and land.** Data center efficiency metrics (PUE, WUE) cannot be compared unless you check the measurement boundary. **PUE has a structural trap: the less efficient the IT equipment, the better it looks.** Finally, this session returns to the question posed in Session 1, "where is the bottleneck?", and gathers the answers of all seven sessions into one.

PRIMARY SOURCES

IBM Think, "What Is an AI Data Center?" · Bloom Energy, "Data Center Cooling for Hyperscale and AI Workloads" · Bloom Energy 2026 Data Center Power Report: Mid-Year Pulse · Counterpoint, "Global DRAM and HBM Market Share" · KISTEP, "K-Cloud Technology Development Project Using AI Semiconductors" (in Korean, title translated from Korean; ScienceON TRKO202500001876) · IEA, Key Questions on Energy and AI (2026-04) · LBNL press release on its 2024 U.S. data center energy report · Uptime Institute Global Data Center Survey 2025 · NVIDIA technical blog on 800 VDC

Learning objectives

1. Explain what is fundamentally different about an AI data center compared with a conventional one.
2. Describe the power chain from the grid to the transistor and understand the losses at each stage.
3. Use heat transfer to explain why liquid cooling becomes unavoidable above a certain density.
4. Read efficiency metrics such as PUE and WUE critically.
5. Make rough estimates of power, cooling and memory demand from a rack specification.

0. Self-study guide — for readers working alone

> **THIS IS THE SESSION WHERE EVERYTHING LANDS**
>
> The tools built up over six sessions come together here into a single facility. In particular, **Exercise 5 packs the content of all seven sessions into one problem. If you are short of time, skip the other exercises if you must, but do Exercise 5.** That one problem is the final check of this book.

What this session brings in from earlier ones — it uses all of them

FROM	WHAT	WHERE IT IS USED
Session 2	Decoding ceiling formula	Exercise 6 (power cost per token)
Session 3	HBM capacity and stack structure	Exercise 5 (c)
Session 4	Yield and supply constraints	Macro figures in section 6
Session 5	BSPDN (backside power delivery)	Exercise 4 (IR drop)
Session 6	Thermal resistance chain $R = t\ /\ (kA)$	Exercise 1 and section 3 of this session

Prerequisite check

CONCEPT	ONE-LINE ANSWER
Specific heat c_p	Energy needed to raise 1 kg of a substance by 1 K. Air about 1,005, water about 4,186 J/(kg·K)
Mass flow rate $\dot{m}$	Mass of fluid flowing per second [kg/s]
$Q = \dot{m}c_p\Delta T$	The heat a fluid can carry per unit time, that is, the heat transfer rate [W]. **The formula used most in this session**
Ohm's law $V = IR$	Used as is in Exercise 4
kW and kWh	Instantaneous power versus accumulated energy. 1 kW for 1 hour = 1 kWh
PUE	Total facility power ÷ IT equipment power. 1.0 is the ideal

Reading order (135 minutes recommended)

STEP	SECTION	TIME	NOTE
1	Sections 1–2 (data center structure, power chain)	25 min	Do Exercise 2 here
2	**Section 3, cooling and the thermal resistance chain**	25 min	An extension of Exercise 4 in Session 6. Do Exercise 1 here
3	**Section 4, PUE and its traps**	20 min	**Be sure to work Exercise 3.** It is training in criticizing a metric
4	Sections 5–7: networks, macro figures, the memory market	20 min	Closes the loop opened in section 4 of Session 1
5	**Exercise 5**	**30 min**	**The final synthesis.** Write everything out on a single sheet of paper
6	Exercises 4 and 6 + self-assessment	15 min	

One sentence that runs through this session

> **All power ends up as heat, and all heat has to go somewhere.** The ceiling of a data center is set by these two sentences, not by its computing capacity.

Where readers get stuck — "why water, all of a sudden?"

Why a rise in rack power from 10 kW to 120 kW forces a change of cooling method is explained by one difference in the properties of water and air.

	DENSITY	SPECIFIC HEAT	HEAT CAPACITY PER VOLUME
Air	1.18 kg/m³	1,005 J/(kg·K)	1,186 J/(m³·K)
Water	997 kg/m³	4,186 J/(kg·K)	4,173,000 J/(m³·K)

$$\frac{4{,}173{,}000}{1{,}186} \approx 3{,}500\times$$

To carry the same heat, water needs 1/3,500 of the volume that air does. Removing 120 kW with air means pushing 6.75 m³ per second (about one small room's worth, roughly 14,300 CFM) through a 1.2 m² rack face, which gives an average face velocity of about 5.6 m/s. That airflow is about 7.5 times what a single top-performing floor tile delivers (1,900 CFM, see section 3). The fan power, noise and pressure loss of forcing that much air through the servers grow with it. Exercise 1 checks this in numbers.

> **In one line:** liquid cooling was adopted not because it is a better technology but because **at this density the airflow, fan power and noise that air cooling needs exceed the practical range.**

---

1. Getting the intuition — when density changes, everything changes

What changed

A conventional data center filled each rack with servers drawing a few kilowatts and cooled them with air. An AI rack is denser by an order of magnitude or more.

ITEM	CONVENTIONAL DATA CENTER	AI DATA CENTER
Rack power density	5–10 kW class	**tens of kW to over 100 kW** (the latest systems go higher)
Cooling	Air cooling (CRAC/CRAH)	**Liquid cooling required** (direct-to-chip, immersion)
Load profile	Relatively smooth	**Synchronized swings of training jobs** — thousands of machines ramp load up and down together
Network	Mainly north-south traffic	**Explosion of east-west traffic** (collective communication between nodes)
Siting constraint	Latency (closeness to users)	**Whether power can be secured**
Bottleneck	Land and buildings	**Grid connection**

> **In one sentence:** the binding question for an AI data center has changed from "where would we like to build?" to **"where can we draw power from?"**

The gap between two worlds (as of October 2026). Across facilities as a whole, density is still low. Uptime Institute's 2025 survey (published July 2025) reports that "Average server rack power densities continue to rise slowly, driven by greater adoption of racks in the 10 kW to 30 kW range. Few facilities exceed 30 kW, and extreme densities remain rare." The people building AI racks see ten times that. In a May 2025 technical blog post, NVIDIA wrote that "As racks exceed 200 kilowatts," today's 54 V in-rack distribution "begins to hit physical limits" and announced a move to 800 V direct-current distribution "to support 1 MW IT racks and beyond, starting in 2027" (section 2). The "over 100 kW" in the table above refers to the AI side of these two worlds.

Why density went up — continued from Session 6

This is not an arbitrary change but a direct consequence of the earlier sessions.

1. **Compute and memory must sit close together to perform** (Session 2, Session 3, Session 6) → GPU and HBM are integrated in one package → power per package reaches the kilowatt class
2. **Communication between nodes is a bottleneck** → nodes must be physically close for high-bandwidth links (NVLink and the like) → they are packed into the rack
3. **Result:** GPUs that must form one group over high-bandwidth links (the scale-up domain of section 5) perform poorly if spread out, so within that group density cannot be lowered. **Within that scope, density is not a choice but a requirement of the**

architecture. How large a group to put in one rack remains a design branch that depends on the performance target (8 versus 72 accelerators per rack in Exercise 5 (d)).

---

2. The power chain — from the grid to the transistor

```
[Transmission grid (tens to hundreds of thousands of V)]
        ↓ receiving substation
[Medium-voltage distribution (thousands to tens of thousands of V)]
        ↓ transformer
[Low-voltage distribution (hundreds of V)]
        ↓ UPS (uninterruptible power supply) + generator backup
[PDU / busbar (rack distribution)]
        ↓
[Rack power (trending toward DC distribution)]
        ↓ PSU
[Board power]
        ↓ VRM (voltage regulator module)
[Chip (1 V or less, hundreds to thousands of A)]
        ↓ package power network → on-die power network
[Transistors]
```

Losses multiply at each stage

If there are several stages with efficiency η_i, the overall efficiency is $\prod_i \eta_i$. Even at 97% per stage, five stages give 86%. **Every loss becomes heat that must in turn be cooled.** That is why better distribution efficiency pays twice: it also lightens the cooling load.

This is why **high-voltage direct current (HVDC) distribution** and **rack-level power integration** are being pursued. Fewer conversion stages mean less loss and less heat.

800 V DC — when one rack becomes a megawatt. Inside today's AI rack, power shelves produce direct current and copper busbars carry it to the compute trays. This is commonly called "48 V distribution" (48 V is the nominal voltage of the industry standard OCP Open Rack V3). NVIDIA's documents give the in-rack bus voltage as 54 V. This session groups the two together as the 48 V-class system. In its May 2025 technical blog post, NVIDIA judged that this approach hits its limits once a rack exceeds 200 kW and announced a move to **800 V direct current (800 VDC) distribution supporting IT racks of 1 MW and more, starting in 2027.** The same post states that "The physics of using 54 VDC in a single 1 MW rack requires up to 200 kg of copper busbar." Raising the voltage lowers the current for the same power, and resistive loss falls with the square of the current (the solution to Exercise 4 shows this in numbers). The same post claims that the design "Improves end-to-end efficiency by up to 5%" and can cut "total cost of ownership (TCO) by up to 30%". These are **NVIDIA's claims**, not independently verified values.

The last centimeter is the hardest — the link to Session 5

A chip must be supplied with **hundreds to thousands of amperes at 1 V or less.** In $P = VI$, a small V means a large I.

The problem: the wiring resistance R produces a voltage drop $\Delta V = IR$ (IR drop). With a large

current this drop is no longer negligible, and transistors in the middle of the chip fail to receive their specified voltage. The resistive loss $P_{\text{loss}} = I^2R$ also grows with the square of I .

Remedies

- Devote many metal layers to the on-die power delivery network (PDN) → it competes with signal wiring for space
- Absorb sudden load swings with decoupling capacitors
- **Backside power delivery (BSPDN)** — the one we met in Session 5. Moving the power network to the back of the wafer relieves congestion on the front and reduces IR drop

> ★ **The book's loop closes here.** The power problem of the data center changes the structure of the transistor (BSPDN), and that change in turn raises the difficulty of the process (wafer thinning, alignment). **This session and Session 5 are each other's cause and effect.**

---

3. The physics of cooling — why air is not enough

The basic formula

The heat a cooling fluid can take away per unit time, that is, the heat transfer rate Q [W], is

$$Q = \dot{m}\, c_p\, \Delta T$$

- $\dot{m}$: mass flow rate [kg/s]
- c_p : specific heat [J/(kg·K)]
- ΔT : temperature difference between fluid inlet and outlet

Multiplying the units gives kg/s × J/(kg·K) × K = J/s = W. The Q in this session is not thermal energy [J] but the rate at which heat flows.

The heat transfer rate from a surface into the fluid is

$$Q = hA\Delta T_{\text{surface}}$$

- h : convective heat transfer coefficient [W/(m²·K)]
- A : heat transfer area

Air vs. water

PROPERTY	AIR	WATER	RATIO
Density (kg/m^3)	~1.2	~1000	**about 830×**
Specific heat c_p (J/kg·K)	~1005	~4180	about 4×
Volumetric heat capacity ρc_p (kJ/m^3·K)	~1.2	~4180	**about 3,500×**
Thermal conductivity (W/m·K)	~0.026	~0.6	about 23×
Convective heat transfer coefficient h (typical)	10–100	1,000–10,000	**10–100×**

For the same volume flow, water carries about 3,500 times as much heat as air. Cooling 100 kW with air needs an unrealistic airflow, and the fan power to produce that airflow becomes enormous in its own right (fan power scales roughly with the cube of airflow).

> **Quick exercise:** to air-cool a single 30 kW rack with an inlet-outlet temperature difference of 15 K, what airflow is needed? $\dot{V} = Q \,/\, (\rho c_p \Delta T) = 30000 \,/\, (1.2 \times 1005 \times 15) \approx 1.66$ m^3/s ≈ 3,500 CFM. At 100 kW, one rack needs more than three times that. **Given the area of a rack's front face, the air velocity becomes unrealistic.**

The spectrum of cooling methods

METHOD	APPLICABLE DENSITY (ROUGHLY)	PRINCIPLE	PROS AND CONS
Air cooling (CRAC/CRAH)	Roughly 30 kW/rack (depends on conditions)	Separate cold and hot aisles, fan circulation	Simple and cheap. Clear density limit
Rear-door heat exchanger (RDHx)	~40 kW/rack	Water-cooled coil in the rack's rear door	Can be used alongside existing air-cooling infrastructure
Direct-to-chip cooling (D2C, cold plate; also called direct-to-chip liquid cooling)	tens of kW to 175 kW/rack	A cold plate on top of the chip circulates coolant	**The de facto standard for AI racks.** Needs CDUs, leak detection and piping. Components other than the chip still need air cooling
Single-phase immersion cooling	High	Servers submerged in a dielectric liquid	No fans, uniform cooling. Issues with maintenance, fluid cost and compatibility
Two-phase (boiling) immersion cooling	Very high	Liquid absorbs heat as latent heat while boiling	Best at handling heat flux. Fluid regulation (PFAS etc.) and system complexity

Status in 2026: in high-density AI racks, direct-to-chip cooling has become the de facto standard, and immersion cooling is used only in limited, specific applications.

Why the air-cooling limit is not pinned to one number. The ceiling of air cooling depends on how air is supplied (raised-floor tiles, hot-aisle containment, in-row coolers) and on the allowed temperature difference, so sources differ. The 2021 white paper of ASHRAE TC 9.9 states: "Air flows of a 40 to 50 kW rack could be up to 5000 cfm. Floor tile best-in-class is 1900

cfm." Redoing the quick exercise above with an air density of 1.18 kg/m^3, removing 40 kW at $\Delta T = 15$ K takes about 4,760 CFM, which agrees with that figure. Even the quick exercise's 30 kW (about 3,500 CFM) is more than one tile can supply. For such racks the white paper says designers must check whether the number of floor tiles the aisle space allows would actually deliver enough airflow, and it names supplemental cooling such as overhead or in-row cooling and rear-door heat exchangers as alternatives. This session therefore reads the air-cooling limit as **roughly 30 kW per rack with floor-tile supply alone.** A 40–50 kW rack needs supplemental cooling unless the layout lets several tiles supply the required airflow. The concentration of field deployments below 30 kW in the Uptime survey in section 1 reflects deployment status, not a physical limit, but it points the same way.

Cold water as a place to dump heat — water-source cooling. Instead of cooling towers, some sites draw cold water from deep in a reservoir behind a dam or from a deep lake or the sea and use a heat exchanger to cool the coolant. The principle is the same formula, $Q = \dot{m} c_p \Delta T$. With assumed values, drawing 1 m^3/s of cold water (about 1,000 kg/s) and returning it only 5 K warmer carries $1{,}000 \times 4{,}186 \times 5 \approx 20.9$ MW. Removing the 15.6 MW facility power of the 10,000-accelerator cluster in Exercise 5 under these conditions takes, by calculation, about 745 kg/s. If the water stays cold and steady all year, the chiller load falls. In exchange come intake and discharge permits and the ecological effect of warmer discharge water. Trap 4 in section 4 (trading water for electricity) applies here too.

The link to Session 6 — heat was a problem inside the chip from the start

The problem in 3D stacking of heat from the lower chip having to pass through the upper chip (Session 6, section 7) becomes here the problem of heat passing from the chip through the cold plate into the coolant, and on to the CDU and the cooling tower. **The thermal resistances form one chain in series**, and a large resistance anywhere blocks the whole chain.

$$\Delta T_{\text{total}} = Q \times (R_{\text{junction-case}} + R_{\text{TIM}} + R_{\text{coldplate}} + R_{\text{fluid}} + \cdots)$$

This is why the **TIM (thermal interface material)** matters more than you would expect. The tiny gap between chip and cold plate can become the bottleneck of the entire chain.

---

4. Reading efficiency metrics critically

Definitions

$$\text{PUE} = \frac{\text{total data center power}}{\text{IT equipment power}}$$

The ideal value is 1.0. The larger the value, the more the supporting systems such as cooling and power distribution consume.

$$\text{WUE} = \frac{\text{annual water use [L]}}{\text{IT equipment energy [kWh]}}, \qquad \text{CUE} = \frac{\text{total } CO_2 \text{ emissions}}{\text{IT energy}}$$

The structural traps of PUE — the core of this section

Trap 1: the denominator is IT power. If the supporting systems carry a fixed load unrelated to IT power, then the more inefficient the servers, the larger the denominator and **the better the PUE** (Exercise 3). At the extreme, adding servers that do no work and only draw power improves PUE. **PUE does not reflect the amount of useful work at all.**

Trap 2: the measurement boundary is arbitrary.

- Are generation losses included? Transmission losses?
- How is on-site generation or on-site renewable power counted?
- Are office space and security systems included?
- At which point is IT power measured (UPS output? PDU? server input?)

Measured at different points, the same facility has different PUEs.

Put two real numbers side by side and the boundary problem appears. In Uptime Institute's 2025 survey, "respondents' annual PUE had a weighted average of 1.54" (681 responding facilities), "marking the sixth consecutive year that this headline figure has virtually stood still." In the same survey, "Facilities commissioned within five years of the survey (from around early 2020) achieved an average PUE of 1.48." Google publishes a 2025 fleet-wide average PUE of 1.09 for its own data centers. It states the boundary: a trailing twelve-month (TTM) average, including all overheads, covering large-scale facilities in stable operation. The difference between the two contains a real difference in efficiency, but one is a weighted average of survey responses and the other is a single company's measurement within a boundary it set itself. Before subtracting the two and reading the result as "how much more efficient Google is," check how their samples and boundaries differ. For reference, putting the two values into the 13 MW IT cluster of Exercise 5 gives a facility power of 20.02 MW at 1.54 and 14.17 MW at 1.09 (by calculation).

Trap 3: timing and conditions. Is it a design PUE or a measured annual average? Which season? At what load factor? **PUE gets much worse at partial load.**

Trap 4: trading water for electricity. Evaporative cooling uses less power but much more water. PUE alone looks better, but WUE gets worse. **Optimizing one metric shifts the cost onto another resource.**

> **A frame for readers trained in statistics:** PUE is a textbook case of **Goodhart's law** — when a measure becomes a target, it ceases to function as a measure. The arbitrariness of the measurement boundary is a **problem of comparability.** Comparing values measured with different boundaries is statistically meaningless.

Better metrics?

- **Energy per unit of work** (e.g., kWh per training run, kWh per million inference tokens) — puts useful work in the denominator
- **Hourly matching of carbon intensity** — whether clean power was matched hour by hour, not netted against an annual total
- **Marginal emissions** — what generation this data center actually caused to be added (marginal, not average)

---

5. Networks — scale-up and scale-out

Two axes

	SCALE-UP	SCALE-OUT
Scope	Inside a node/rack	Between nodes, between racks
Technology	NVLink, proprietary interconnects	InfiniBand, Ethernet (RoCE)
Bandwidth	Very high	Relatively low
Latency	Very low	High
Scaling limit	Physical proximity (tens of GPUs)	Tens of thousands

How parallelization strategies map onto them (the part familiar to AI readers)

- **Tensor parallelism (TP):** splits one layer across several GPUs. Needs an all-reduce at every layer → **the traffic is enormous, so it must stay inside the scale-up domain**
- **Pipeline parallelism (PP):** places groups of layers on different GPUs. Little traffic → can scale out. In exchange, bubbles (idle time) appear
- **Data parallelism (DP):** splits the batch. A gradient all-reduce at the end of each step → communication can overlap with computation
- **Expert parallelism (EP, MoE):** all-to-all communication → the irregular pattern makes network design hard

> **Why this decides data center design:** tensor parallelism requires GPUs to be physically close. So they are packed into the rack as densely as possible, which raises rack power density, which in turn requires liquid cooling. **It is a causal chain in which the model's parallelization strategy decides the design of the building.**

Where collective communication becomes the bottleneck

In a ring all-reduce over N GPUs, each GPU sends and receives roughly $2\frac{N-1}{N}\times$ (parameter size), which converges as N grows. The problem is that **latency** grows in proportion to the number of hops. That is why large systems use hierarchical all-reduce (within the rack first, then between racks).

Optical interconnect (CPO — co-packaged optics)

Electrical signals lose more and burn more power the farther they travel (Session 6, section 9). Light loses far less with distance.

The idea of CPO: put the optical transceiver **in the same package** as the switch chip so that electrical signals travel as short a distance as possible. This can cut power substantially.

Challenges: temperature sensitivity of optical devices, laser reliability, serviceability (once inside the package, parts cannot be replaced individually), and cost. **This is where the packaging technology of Session 6 extends into the network.**

How far commercialization has come (as of October 2026, from manufacturer announcements only). On October 8, 2025, Broadcom announced that it had begun shipping its third-generation CPO switch (Tomahawk 6 – Davisson), which packages optical engines together with a 102.4 Tb/s Ethernet switch chip. The 70% reduction in optical-interconnect power compared with conventional pluggable modules is the company's claim. In March 2025, NVIDIA announced the Quantum-X Photonics CPO switch for InfiniBand and the Spectrum-X Photonics CPO switch for Ethernet, with the former slated for later in 2025 and the latter for 2026. Whether shipments have actually begun could not be confirmed from an official NVIDIA announcement as of October 2026. Both companies' announcements concern switches with attached optical engines.

---

6. Macro figures and their uncertainty

Why the numbers diverge so much

Estimates of data center power consumption vary widely by institution, year and definition. Taking one institutional outlook as a reference point: the International Energy Agency's *Key Questions on Energy and AI* (April 2026) sees global data center electricity consumption "roughly doubling from 485 TWh in 2025 to 950 TWh in 2030, accounting for around 3% of global electricity demand by that date" (its reference scenario, the Base Case). Growth in 2025 alone was 17%. For the United States alone, a December 2024 report by Lawrence Berkeley National Laboratory (LBNL), supported by the U.S. Department of Energy, estimates 176 TWh in 2023 (4.4% of U.S. electricity) and 325–580 TWh in 2028 (6.7–12%). That the 2028 value comes as a range nearly two-fold wide is itself the size of the uncertainty. Misconception 6 in section 6 of Session 1 uses the same figures to separate efficiency from total volume. To add one sense of scale: a 1 GW power plant running at a 90% capacity factor produces about 7.9 TWh a year, by calculation. Figures from private research firms, however, differ considerably from these.

Sources of the differences

1. **Whether cryptocurrency mining is included** — this alone moves tens to hundreds of TWh
2. **Scope** — commercial colocation only, or corporate in-house server rooms too? Telecom facilities?
3. **AI-only vs. total** — how is an "AI data center" distinguished? What about mixed-workload facilities?
4. **Estimation method** — estimating from accelerators shipped × rated power × utilization is highly sensitive to the assumptions
5. **Publication date** — outlooks are revised substantially every six months. Even without a revision, the number that gets quoted changes. The IEA's 2025 *Energy and AI* projected a path from about 415 TWh in 2024 to about 945 TWh in 2030; the edition a year later still describes its Base Case in the main text as "growing from around 415 TWh in 2024 to around 950 TWh in 2030" while its summary starts from the 485 TWh of 2025: "roughly doubling from 485 TWh in 2025 to 950 TWh in 2030" (the annex table gives 416 TWh for 2024, 485 TWh for 2025 and 945 TWh for 2030). The IEA itself says its central projection "remains close to the trajectory set out in the IEA's 2025 report." What changed is not the 2030 outlook but which year is quoted as the starting point, and the rounding. Before putting the two numbers side by side and reading "the outlook went up," align the base years
6. **Interests of the publisher** — regulatory debates, fundraising and environmental campaigns each carry different incentives

> **The lesson of Session 1 repeats.** What this book settles is **trends and structure**, not particular numbers. What can be settled: ① data center power consumption is rising fast, ② the main driver of that rise is AI workloads, ③ rack power density is rising steeply, ④ as a result, liquid cooling has become standard.

The power grid as the new bottleneck

Even with chip and packaging capacity secured, **the grid connection queue** becomes the bottleneck. New transmission lines and grid interconnection approvals take years. The IEA's April 2026 report states that "Waiting times for grid connections can be as long as five to ten years in many jurisdictions." It also writes that in the United States slow connections are leading more data center projects to add on-site gas generation. Hence:

- Preference for sites near power plants
- Consideration of self-generation (gas turbines, fuel cells)
- Interest in nuclear power (power purchase agreements with existing plants, longer-term plans for SMRs)
- Flexible load (demand response) — the idea that training workloads, being less sensitive to latency, can be adjusted to the grid's headroom

This book does not cover the economic, policy and legal issues at this point.

7. Back to the memory market — the last link

The book returns to the memory market in its last session because this is where the loop closes.

The causal chain

```
AI data center build-out
      ↓ more demand for accelerators
Each accelerator needs several HBM stacks
      ↓
HBM demand surges
      ↓ HBM consumes far more die area than ordinary DRAM
      │  (more wafers are needed for the same number of bits)
Supply of ordinary DRAM is crowded out
      ↓
Prices rise across memory
      ↓
Data center build-out costs rise in turn
```

The core mechanism: from the same wafer capacity, HBM production yields far fewer bits than ordinary DRAM (TSV area, stack yield losses, larger dies). In other words, **shifting capacity to HBM reduces the total supply of DRAM bits.** This is the path by which AI demand spills over into the memory market as a whole.

Final discussion — does the cycle repeat?

Arguments for "this time is different"

- Inference demand, unlike training, is **continuous and an operating expense.** A service, once deployed, keeps running
- HBM volumes are fixed under long-term contracts with customers, unlike the traditional spot market
- GPU leasing, too, is locked into long-term commitments. In the fiscal 2025 annual report (10-K) of the GPU cloud company CoreWeave, committed contracts are "typically structured as 'take-or-pay' agreements, requiring payment regardless of the level of utilization." Committed contracts accounted for 88% of revenue in 2023, 96% in 2024 and over 98% in 2025. The weighted-average contract duration is approximately five years, and the company uses these commitments as collateral to raise asset-level debt and buy equipment
- Barriers to entry are much higher (advanced packaging, stacking technology)

Arguments for "it will turn out the same"

- The structure of build-out lead time (2–3 years) lagging behind demand signals has not changed
- All three companies are expanding HBM capacity substantially
- AI capital spending itself may be cyclical. And the plans move within months. In its earnings release of January 28, 2026, Meta put its 2026 "capital expenditures, including principal payments on finance leases," at USD 115–135 billion, and in its July 29 release it changed the range to USD 130–145 billion (the July release describes this as "narrowed from our prior outlook" of USD 125–145 billion). Taking the midpoints of the ranges, it rose 10% in half a year. A number that moves this fast on the way up can move just as fast on the way down
- In the past, too, there was always a claim that "this time the demand is structural"

If you read this as a group: split into two teams and have them debate. The content of all seven sessions comes into play naturally. Yield (Session 4), the cost structure of HBM (Session 3), packaging capacity (Session 6) and power constraints (this session) all become arguments.

---

8. Common misconceptions

Misconception 1: "A low PUE means environmentally friendly" → PUE looks only at the efficiency of the supporting systems. It reflects neither the amount of useful work, nor the source of the power (coal or renewables), nor water use.

Misconception 2: "Liquid cooling is a risky, exotic technology" → In high-density AI racks it is already the standard. Water cooling was used in the mainframe era as well.

Misconception 3: "Build a data center in a cold region and cooling is free" → Free cooling with outside air helps, but in high-density racks the thermal resistance from chip to coolant dominates, so outside temperature alone does not solve the problem. There are also issues of the power grid, network latency and staffing.

Misconception 4: "Secure the power and you can build a data center" → Grid connection approval, site permits, water supply and local acceptance are all constraints.

Misconception 5: "AI data centers will bring down the grid / their impact is negligible" → Both are exaggerations. In total they are a fraction of all electricity, but **concentrated in one area, they are a large shock to that area's grid.** Distinguish the total from the local impact.

Misconception 6: "If HBM sells well, every memory company does well" → Converting capacity to HBM reduces the supply of ordinary DRAM. The structure of the whole market changes, and winners and losers separate.

---

9. Quantitative exercises

Exercise 1 — Deriving cooling requirements from a rack specification For a rack drawing 120 kW: (a) What airflow is needed to hold $\Delta T = 15$K with air cooling? Assuming a rack front area of 0.6×2m, what is the air velocity? Is it realistic? (b) What coolant flow is needed to hold $\Delta T = 10$K with water cooling? (L/min)

Exercise 2 — Losses in the power chain With transformer (99%) → UPS (96%) → PDU (99%) → PSU (94%) → VRM (90%), what is the overall efficiency from the grid to the chip? To supply 1 MW of IT power, how much must be drawn from the grid? Since every loss becomes heat, how does it add to the cooling load?

Exercise 3 — Calculating and gaming PUE What is the PUE of a facility with 10 MW of IT power, 3 MW of cooling and 1 MW of distribution losses? Now if the IT equipment is made more efficient and does the same work with 8 MW, what happens to PUE? Work the two cases separately. (A) Cooling and distribution losses fall in full proportion to IT power. (B) Of the 3 MW of cooling, 1 MW, and of the 1 MW of distribution losses, 0.2 MW, are fixed loads unrelated to IT power, and only the rest falls in proportion. **Confirm that the work is the same, yet PUE stays the same or even gets worse.**

Exercise 4 — IR drop Supplying 1V and 500A to a chip, what is the voltage drop if the equivalent resistance of the power network is 0.2 mΩ? What voltage actually reaches the core? What would you have to do to halve this drop?

Exercise 5 — Estimating the scale of a data center (synthesis) Assume a training cluster of 10,000 accelerators and estimate the following. (a) Total IT power (assume 1 kW per accelerator + 30% for supporting equipment) (b) Total facility power and annual energy at a PUE of 1.2 (c) Total HBM capacity and number of stacks required (d) Number of racks and rack power density at 8 accelerators per rack, and the cooling method required **The content of all seven sessions enters this single problem.**

Exercise 6 — Cost of an inference service Combine the decoding-ceiling formula from Session 2 with the power figures above to make a rough estimate of the power cost of generating one million tokens. How does it change as the batch size changes?

10. Discussion questions

1. Is the real bottleneck of AI data centers the chip, packaging, memory or power? How has it moved over time, and where does it go next?
2. If you were to design a metric to replace PUE, what would you put in the numerator and denominator? How would you prevent it from being gamed?
3. Can the flexible-load character of training workloads be used to stabilize the grid? What conditions would that require?
4. If density keeps rising, where is the physical limit of cooling? When that limit is reached, how must the architecture change?
5. Is the memory cycle different this time? (See section 7 — make the case for both sides before you judge.)
6. **Wrap-up:** on the value-chain map you drew in Session 1, mark the bottlenecks learned over the seven sessions. Where would putting resources be most effective?

---

11. Further study

- **Data center infrastructure:** ASHRAE data center resources (Datacom Series) — the industry standard for temperature and humidity guidelines and cooling design. For the background to the shift to liquid cooling, see the TC 9.9 white paper "Emergence and Expansion of Liquid Cooling in Mainstream Data Centers" (2021)
- **Efficiency metrics:** The Green Grid's white papers on PUE and WUE (definition documents from the body that created the metrics; e.g., "PUE: A Comprehensive Examination of the Metric"). Uptime Institute's annual data center survey (2025 edition)
- **Energy outlook:** IEA, *Energy and AI* (2025) and its follow-up *Key Questions on Energy and AI* (2026-04) — the most widely cited institutional outlook. **Be sure to read the methodology chapter alongside it.** The U.S. data center energy report (2024-12) of LBNL (Lawrence Berkeley National Laboratory) is worth consulting for its transparent methodology
- **Networks and systems:** Barroso, Hölzle & Ranganathan, *The Datacenter as a Computer* (3rd ed.) — **the standard textbook on warehouse-scale computing. Especially recommended for readers with a computing background**
- **Communication in distributed training:** the parallelization sections of the Megatron-LM (arXiv:1909.08053) and DeepSpeed ZeRO (arXiv:1910.02054) papers
- **Optical interconnect:** CPO talks at OFC (Optical Fiber Communication Conference)

Self-study appendix A — Solutions to the quantitative exercises

Solution to Exercise 1 — Deriving cooling requirements from a rack specification

▾ Solution

(a) Air cooling, 120 kW, $\Delta T = 15$ K

$$\dot{m} = \frac{Q}{c_p \Delta T} = \frac{120{,}000}{1{,}005 \times 15} = 7.96 \text{ kg/s}$$

Converting with an air density of 1.18 kg/m³ (about 25 °C):

$$\dot{V} = \frac{7.96}{1.18} = 6.75 \text{ m}^3/\text{s} = 405 \text{ m}^3/\text{min} \approx 14{,}300 \text{ CFM}$$

Air velocity across the rack front of $0.6 \times 2 = 1.2$ m²:

$$v = \frac{6.75}{1.2} = 5.6 \text{ m/s}$$

Is it realistic? No. Three problems bite at once.

PROBLEM	DETAIL
Noise	Fan noise scales roughly with the 5th to 6th power of air velocity. Doubling the velocity with the same fan makes it tens of times louder (2^5–2^6 = 32–64 times)
Fan power	Fan power consumption scales with the cube of the flow. The power spent on cooling turns back into heat, a vicious circle
Pressure loss	Pushing air through the server at that speed sends the static pressure soaring, and recirculation and bypass between front and rear racks get worse

The practical limit differs by source and conditions, but with floor-tile supply alone it is roughly 30 kW per rack (section 3, "Why the air-cooling limit is not pinned to one number"). According to the ASHRAE TC 9.9 white paper, the airflow of a 40–50 kW rack is up to 5,000 CFM, 2.6 times one best-in-class floor tile (1,900 CFM). The roughly 14,300 CFM of a 120 kW rack is about 7.5 times one tile and about 2.9 times the maximum airflow of a 40–50 kW rack, so cooling it with floor-tile supply is outside the practical range.

(b) Water cooling, $\Delta T = 10$ K

$$\dot{m} = \frac{120{,}000}{4{,}186 \times 10} = 2.867 \text{ kg/s} \approx 2.87 \text{ L/s} = 172 \text{ L/min}$$

Comparing the volume flow of the two methods

METHOD	VOLUME FLOW
Air cooling	6.75 m^3/s
Water cooling	0.00287 m^3/s
Ratio	**about 2,350×**

With a 40 mm pipe diameter, the flow velocity is about 2.3 m/s, within the recommended range (1–3 m/s). Narrowing it to 25 mm gives 5.8 m/s, which brings erosion and noise problems.

What to take away: the change of cooling method is **a result forced by material properties.** And it changes the design of the whole data center — CDUs, secondary piping loops, leak detection and maintenance procedures are all needed anew.

The figure below shows a 120 kW rack cooled by air at $\Delta T = 15$ K (air density 1.18 kg/m^3) and by water at $\Delta T = 10$ K. In the screen edition you can change the rack power, the two temperature differences, the air density, the rack front area and the pipe diameter, and see how airflow, air velocity and coolant flow change against the two ASHRAE white paper reference points (5,000 and 1,900 CFM).



Figure 7-1. Cooling a 120 kW rack with air (ΔT 15 K) takes 6.75 m^3/s (about 14,300 CFM) and an air velocity of 5.6 m/s across the rack face. Liquid cooling (ΔT 10 K) needs only 172 L/min. The two dashed lines are the 5,000 and 1,900 CFM figures from the ASHRAE white paper. — In the HTML edition you can change the values here (HTML edition, Session 7, Solution to Exercise 1).

Solution to Exercise 2 — Losses in the power chain

▾ Solution

Overall efficiency

$$0.99 \times 0.96 \times 0.99 \times 0.94 \times 0.90 = 0.796$$

Step by step: $0.99 \times 0.96 = 0.9504 \rightarrow \times 0.99 = 0.9409 \rightarrow \times 0.94 = 0.8844 \rightarrow \times 0.90 = 0.7960$.

To deliver 1 MW to the chips

$$\frac{1}{0.796} = 1.256 \text{ MW}$$

1.256 MW must be drawn from the grid. **All 256 kW of loss becomes heat.**

Where this heat is added — location matters

LOSS POINT	EFFICIENCY	LOSS	WHERE THE HEAT COMES OUT
Transformer	99%	12.6 kW	Electrical room
UPS	96%	49.7 kW	Electrical room
PDU	99%	11.9 kW	Around the computer room
PSU	94%	70.9 kW	**Inside the rack**
VRM	90%	111.1 kW	**On the board, next to the chip**

The heat to be removed inside the rack is not the chips' 1 MW but

$$\frac{1}{0.94 \times 0.90} = 1.182 \text{ MW}$$

That is, 18% more. The PSU loss of 70.9 kW and the VRM loss of 111.1 kW come out inside the rack, so they go into the rack's cooling load and enlarge the rack cooling requirement of Exercise 1 by that much. Which path removes that heat depends on the design. If the VRMs share the cold plate, the coolant takes it away; otherwise it remains part of "components other than the chip still need air cooling" in the table in section 3.

The heat load of the whole facility is 1.256 MW — every bit of power that comes in leaves as heat (computation stores no energy).

Design implication: efficiency gains win twice. They cut the electricity bill, and **they cut the heat to be removed (the cooling load) by the same amount.** Raising VRM efficiency from 90% to 94% cuts VRM loss from 111 kW to about 64 kW, a reduction of about 47 kW (and because the earlier stages carry less power, the power drawn from the grid falls from 1.256 to 1.203 MW, about 53 kW), and the heat the facility must remove falls by the same 53 kW. What falls is the heat load, not the power the cooling

plant itself consumes. How much the cooling plant's consumption falls depends on the chiller's coefficient of performance (COP) and the fixed loads of pumps and fans, and cannot be settled from this exercise's assumptions alone.

The figure below shows 1 MW delivered to the chips through five stages at 99, 96, 99, 94 and 90% efficiency. In the screen edition you can change the efficiency of each stage and the chip power, or set all five stages to 97%, and see how much loss each stage produces and where its heat comes out.

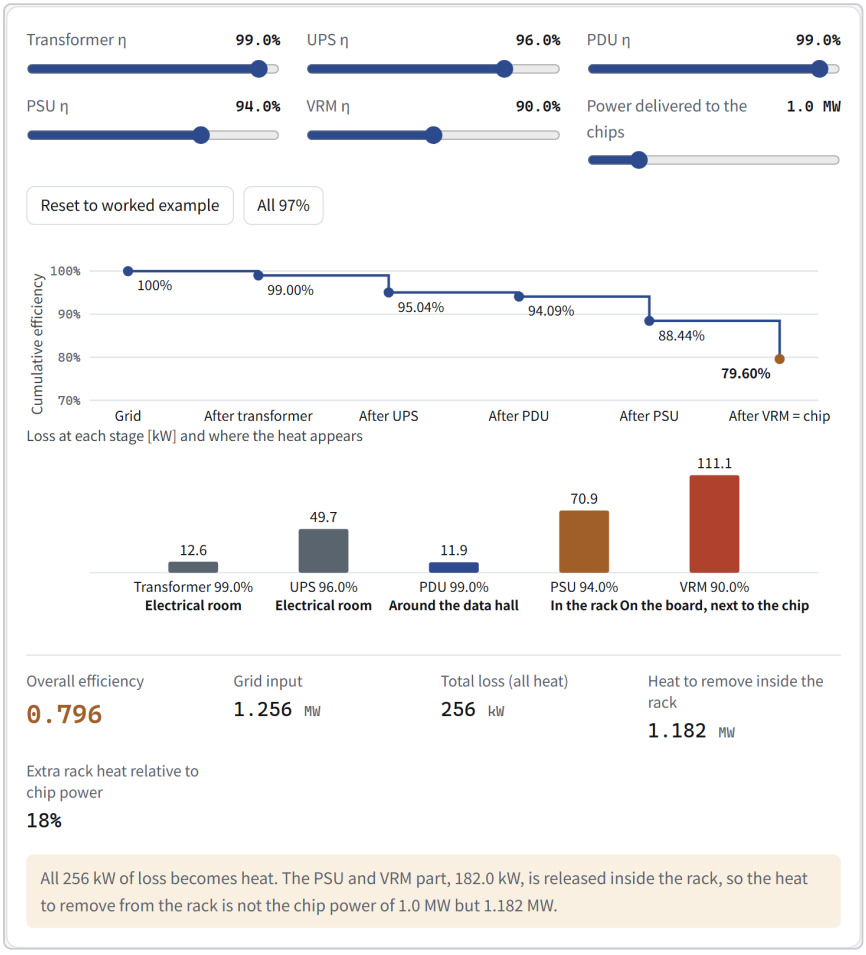


Figure 7-2. With transformer, UPS, PDU, PSU and VRM efficiencies of 99, 96, 99, 94 and 90%, overall efficiency is 0.796, and delivering 1 MW to the chips takes 1.256 MW from the grid. Of the 256 kW lost, the VRM share of 111 kW is the largest. — In the HTML edition you can change the values here (HTML edition, Session 7, Solution to Exercise 2).

Solution to Exercise 3 — Calculating and gaming PUE

▾ Solution

The original facility

$$\text{PUE} = \frac{10 + 3 + 1}{10} = \frac{14}{10} = 1.40$$

(A) IT made efficient at 8 MW — if the supporting systems are fully proportional

Cooling $3 \times 0.8 = 2.4$, distribution $1 \times 0.8 = 0.8$

$$\text{PUE} = \frac{8 + 2.4 + 0.8}{8} = \frac{11.2}{8} = 1.40$$

PUE is unchanged. Total power fell 20%, from 14 to 11.2 MW, yet the metric does not move at all.

(B) In reality it is worse — the supporting systems have fixed loads

Split cooling's 3 MW into 1 MW fixed + 2 MW variable, and distribution's 1 MW into 0.2 fixed + 0.8 variable.

ITEM	IT 10 MW	IT 8 MW
IT	10	8
Cooling	3.0	$1 + 2 \times 0.8 = 2.6$
Distribution	1.0	$0.2 + 0.8 \times 0.8 = 0.84$
Total	**14.0**	**11.44**
PUE	**1.40**	**1.43**

The facility does exactly the same work and cut total power by 18%, yet PUE got worse.

The structural traps of PUE — three of them

1. **The denominator is IT power.** If the supporting systems are fully proportional to IT, making IT more efficient leaves PUE unchanged ((A)); if fixed loads remain, only the denominator shrinks further and the metric gets worse ((B)). **PUE does not measure IT efficiency, and in a facility with fixed loads it even penalizes improvements in IT efficiency.**
2. **The amount of work is absent.** At the same PUE of 1.2, a facility producing ten times the tokens cannot be told apart from one that does not.
3. **It can be gamed by where the boundary is drawn.** The value changes depending on whether transformer losses are counted on the IT side or the supporting side, and whether generation losses are included.

Alternative metrics

METRIC	CONTENT
Energy per unit of computation	J/token, J/FLOP, MWh per training run — **the amount of work goes into the denominator**
Absolute total power	When discussing grid and site constraints, the answer is the absolute amount, not a ratio
TUE (total-power usage effectiveness)	A metric that extends PUE inside the IT equipment. It multiplies ITUE (total IT equipment power ÷ power reaching the computing components inside, which captures PSU, VRM and fan losses) by PUE to give total facility power ÷ power of the computing components. For example, with a PUE of 1.2 and a delivery efficiency of 0.9 inside the IT equipment, ITUE ≈ 1/0.9 and TUE ≈ 1.2/0.9 ≈ 1.33
WUE / CUE	Water and carbon. Reflect local constraints

The lesson when using a metric for judgment: a ratio metric can always be gamed when its denominator is a policy variable. **The habit of first asking "what is the easiest way to make this metric look good?"** — the training that began in section 4 of Session 1 is completed here.

The figure below shows IT of 10 MW (cooling 3 MW = 1 fixed + 2 variable, distribution 1 MW = 0.2 fixed + 0.8 variable) reduced to 8 MW while doing the same work. In the screen edition you can change the fixed and variable parts and the reduced IT power, and compare "fully proportional" with "fixed loads present."



Figure 7-3. Cutting IT load from 10 MW to 8 MW for the same work lowers total power 14 → 11.44 MW, 18% less, but the fixed load of the facility equipment pushes PUE the wrong way, from 1.40 to 1.43. — In the HTML edition you can change the values here (HTML edition, Session 7, Solution to Exercise 3).

Solution to Exercise 4 — IR drop

▾ Solution

Voltage drop

$$\Delta V = IR = 500 \times 0.2 \times 10^{-3} = 0.1 \text{ V}$$

The voltage that actually reaches the core

$$1.0 - 0.1 = 0.9 \text{ V} \quad (10\% \text{ drop})$$

Why it is serious

EFFECT	DETAIL
Slower speed	Transistor drive current is roughly proportional to $(V - V_{th})^{\alpha}$. If V_{th} is 0.3 V, the effective drive voltage falls from 0.7 to 0.6 V, **a 14% decrease**
Collapse of timing margin	The drop is not uniform but varies with position and time (switching noise), so the clock has to be lowered to assume the worst case
Power loss	$I^2R = 500^2 \times 0.2 \times 10^{-3} = 50$ W — if the chip draws 500 W, 10% of it disappears as heat in the power network

Three ways to halve the drop

METHOD	PRINCIPLE	COST
Halve the resistance R	Thicker, wider power wiring, more metal layers	Eats into signal wiring space — competition for front-side routing resources
Backside power delivery (BSPDN)	Moves power wiring to the **back** of the wafer and gives the whole front side to signals. Paths become shorter and thicker, and R falls sharply	Requires new processes (wafer thinning, backside alignment)
Raise the voltage and lower the current	For the same power, doubling V halves I , halves $\Delta V = IR$, and cuts the I^2R loss to **1/4**	The final step-down has to happen close to the chip → on-package voltage regulators (IVR), 48 V DC distribution

The link to Session 5 — why BSPDN matters as much as GAA

Section 3 of Session 5 called backside power delivery only "the next turning point." The reason lies in this calculation. With current rising steadily (V cannot be raised while P rises in $P = VI$, so only I grows), the only way to hold $\Delta V = IR$ in check is to reduce R , and front-side wiring alone has reached its limit. **BSPDN is not a performance-boosting technology but a technology that lifts the physical limit of power delivery.**

And this carries all the way to the data center. Moving rack distribution to 48 V and moving voltage regulators onto the board are both applications of the same formulas ($\Delta V = IR$, loss $= I^2R$) at different levels.

The same calculation at the rack level. This puts the 800 V DC discussion of section 2 into numbers. Feeding a 1 MW IT rack at 54 V, the bus voltage of the 48 V-class system, the current is

$$I_{54} = \frac{1{,}000{,}000}{54} \approx 18{,}519 \text{ A} \approx 18.5 \text{ kA}$$

At 800 V it is

$$I_{800} = \frac{1{,}000{,}000}{800} = 1{,}250 \text{ A}$$

The currents differ by a factor of about 14.8. For conductors of the same resistance, the resistive loss I^2R differs by a factor of

$$\left(\frac{800}{54}\right)^2 \approx 219$$

Put the other way round, matching the 800 V loss at 54 V would require cutting conductor resistance to about 1/219, which for the same length means a cross-section about 219 times larger. NVIDIA's statement that feeding a 1 MW rack at 54 V takes up to 200 kg of copper busbar is the physical form of this calculation. In real designs the conductor size differs by voltage, so the factor of 219 is a comparison that carries the condition "if the same conductor were used."

The figure below covers only the rack-distribution part of this solution. The 0.1 V, 0.9 V and 50 W of the chip IR drop at the start are not in it. It shows a 1 MW IT rack fed at 54 V and at 800 V. In the screen edition you can change the rack power and the two voltages (including the OCP nominal 48 V) and see the currents and the ratio of resistive losses when the same conductor is used.



Figure 7-4. Feeding a 1 MW rack at 54 V takes about 18.5 kA; at 800 V, 1,250 A. The current is 14.8 times higher, so resistive loss in the same conductor is about 219 times higher. — In the HTML edition you can change the values here (HTML edition, Session 7, Solution to Exercise 4).

Solution to Exercise 5 — Estimating the scale of a data center (synthesis)

▾ Solution

This problem is the final check of this book. A training cluster of 10,000 accelerators.

(a) Total IT power

ITEM	CALCULATION	VALUE
Accelerators	$10{,}000 \times 1$ kW	10 MW
Supporting equipment (CPU, storage, network) 30%	10×0.3	3 MW
IT total		**13 MW**

Comparing with real systems. "1 kW per accelerator + 30% supporting" is an assumed value for practice. Switching to a manufacturer's system specification: DGX B200 lists the maximum power of its 8-GPU system as about 14.3 kW. 10,000 accelerators means 1,250 systems, and by calculation a maximum of 17.875 MW. DGX B300, with 8 GPUs and 2.1 TB of memory, is listed at about 14 kW, so the same calculation gives 17.5 MW. Being system-boundary maximums, these exceed the assumed 13 MW. One real installation for comparison: the University of Florida's fourth-generation HiPerGator, with 63 DGX B200 systems (504 GPUs), went into service for the fall 2025 semester. 63 systems times 14.3 kW gives a maximum IT power of about 0.90 MW (by calculation). The GPU portion of a university research supercomputer (63 DGX B200 systems) stays under 1 MW at maximum, and the cluster in this exercise is about 20 times that by GPU count ($10{,}000 \div 504 \approx 19.8$).

(b) Total facility power and annual energy

$$P_{\text{facility}} = 13 \times 1.2 = 15.6 \text{ MW}$$

$$E_{\text{annual}} = 15.6 \text{ MW} \times 8{,}760 \text{ h} = 136{,}656 \text{ MWh} \approx 137 \text{ GWh}$$

Assuming 90% utilization, about 123 GWh.

Getting a feel for the scale: taking a rough 3,000 kWh as the annual electricity consumption of one Korean household,

$$\frac{136{,}656{,}000 \text{ kWh}}{3{,}000} \approx 45{,}600 \text{ households}$$

A single cluster of 10,000 accelerators uses as much electricity as a small city.

> **Comparable cases elsewhere.** Household consumption differs a great deal between countries, so the household count depends on where you stand. The U.S. Energy Information Administration puts the average annual electricity sold to a U.S. residential customer in 2022 at 10,791 kWh; on that basis the same 136,656 MWh corresponds to about 12,700 U.S. households (our calculation) (source).

(c) Total HBM capacity and number of stacks

Taking 192 GB per accelerator (B200 class):

ITEM	CALCULATION	VALUE
Total HBM capacity	10,000 × 192 GB	**1.92 PB**
Number of stacks	8 stacks per accelerator × 10,000	**80,000 stacks**
DRAM dies	8-high per stack → 80,000 × 8	**640,000 dies**

192 GB is the chip's published figure. NVIDIA's Blackwell Architecture Technical Brief (V2.1) gives the GPU memory of HGX B200 as "Up to 192 GB," but the specification of the finished DGX B200 server is 1,440 GB for the eight together, or by calculation 180 GB per GPU (Session 2, section 6, "Capacity also differs between chip and system"). This exercise uses the chip figure, 1.92 PB. Counting by DGX system usable capacity gives 10,000 × 180 GB = 1.80 PB (the same as 1,250 systems × 1,440 GB). In the same way, with DGX B300 (2.1 TB for eight) it is 1,250 systems × 2.1 TB = 2.625 PB (by calculation).

Apply Exercise 1 of Session 3 here. Stacking 8-high without KGD at a die yield of 0.95 gives a stack yield of 66%. Securing 80,000 stacks would require attempting 121,000 stacks, and hundreds of thousands of good dies would be thrown away along the way. This 66% is not an actual yield but the value under the assumption "if dies are not screened before stacking." What the example shows is how much die screening and stack yield can cut supply; this one number cannot explain the actual supply bottleneck behind news that "accelerators can't be built for lack of HBM."

(d) Racks and density — here the design branches

	8 PER RACK (CONVENTIONAL)	72 PER RACK (HIGH DENSITY)
Racks	**1,250 racks**	**139 racks**
Rack power	about 10.4 kW	about 94 kW
Cooling	**Air cooling possible**	**Liquid cooling required** (Exercise 1)
White space (at 4 m^2 per rack)	about 5,000 m^2	about 560 m^2
Communication distance between nodes	Long → larger share of optical transceivers	Short → more spans possible in copper

A single choice of density decides the cooling method, the size of the site and the network cost all at once. The advantage of high density is not saving space but **shortening communication distance** — in training, communication between nodes governs performance, so putting things physically close together is performance. The logic of Exercise 2 in Session 6, "distance sets energy," repeats at rack scale.

(e) The real bottleneck — what appears once you calculate this far

15.6 MW is the incoming power capacity of a mid-sized industrial park. But

ITEM	LEAD TIME
Procuring accelerators	months to 1 year
Constructing the building	1–2 years
New substation and transmission line	**3–7 years**

The lead times in the table are rough orders of magnitude. The IEA's April 2026 report notes that "lead times for transformers average between two and three years" and that the wait for grid connection alone can reach five to ten years in many jurisdictions (section 6). That means it may be longer than the 3–7 years in the table above.

The longest item is the power grid. That is why new data center sites are chosen not by "where the internet is fast" but by **"where power can already be brought in,"** and why self-generation, power purchase agreements (PPAs) and sites next to nuclear plants come up.

Back to the question of Session 1: the bottleneck has moved from EUV → CoWoS → HBM. What this calculation says is that **the next bottleneck lies outside semiconductors, in the power grid.** And neither capital nor technology resolves it quickly.

The figure below uses the assumptions of Exercise 5 (10,000 accelerators, 1 kW each + 30% supporting, PUE 1.2, 192 GB each). In the screen edition you can switch the IT power boundary to the DGX B200 or B300 system maximum, PUE to the 1.54, 1.48 or 1.09 of section 4, and memory per accelerator to the DGX B200 system's 180 GB or to DGX B300. The household count, the share of a 1 GW plant's annual output and the flow for water-source cooling in section 3 move with them.

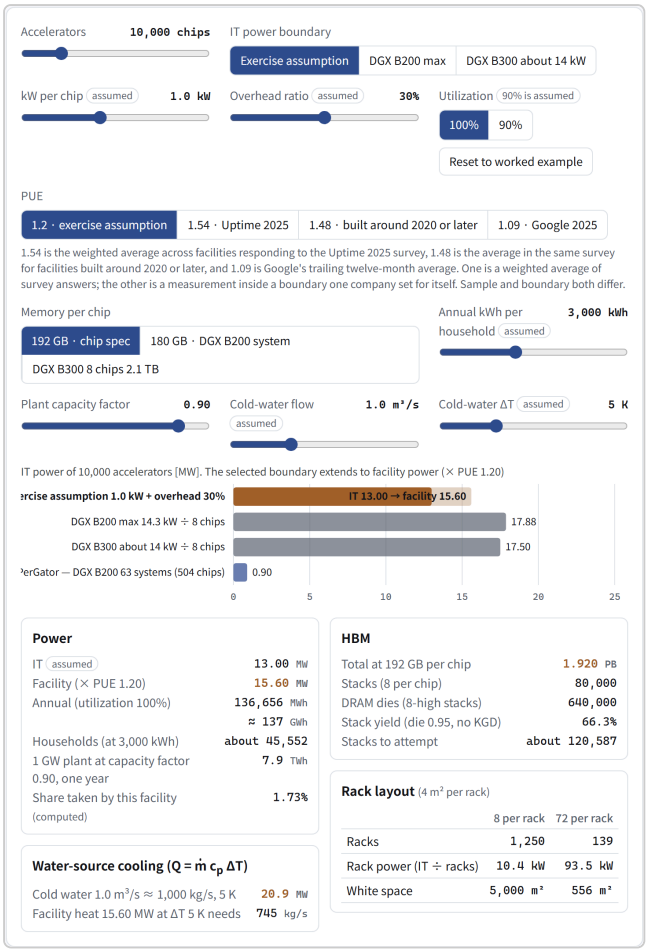


Figure 7-5. 10,000 accelerators (1 kW each plus 30% overhead) mean 13 MW of IT load, 15.6 MW for the facility at PUE 1.2, and about 137 GWh a year. Counted with the DGX B200 and B300 system maximums, IT is 17.875 and 17.5 MW, and at 192 GB per chip HBM totals 1.92 PB. — In the HTML edition you can change the values here (HTML edition, Session 7, Solution to Exercise 5).

Solution to Exercise 6 — Cost of an inference service

▾ Solution

Setup: a 70B INT8 model, an 8×H100 node (total $B = 26.8$ TB/s), node power 6.4 kW, PUE 1.2 → 7.68 kW at the facility level. Electricity at USD 0.08/kWh.

The node power of 6.4 kW is an assumed value for practice. It differs from the manufacturer's specification of 10.2 kW maximum input power for DGX H100. 10.2 kW is a maximum at the system boundary (Exercise 4 of Session 2 distinguishes the chip and server boundaries the same way). Using 10.2 kW makes every power cost below about 1.6 times larger, but the conclusion that capital cost dominates stays the same.

Applying the decoding ceiling from Session 2

$$\text{tokens/s} \lesssim \frac{26.8 \times 10^{12}}{7 \times 10^{10}} \times B_{\text{sz}} = 383 \times B_{\text{sz}}$$

Setting actual bandwidth utilization at 60% gives an effective $230 \times B_{sz}$ tok/s.

Cost of generating one million tokens, by batch

BATCH	EFFECTIVE THROUGHPUT	TIME	ENERGY	POWER COST
1	230 tok/s	4,353 s (1.21 h)	9.29 kWh	**USD 0.743**
8	1,840	544 s	1.16 kWh	**USD 0.093**
32	7,350	136 s (0.038 h)	0.290 kWh	**USD 0.023**

Batch 32 is exactly 32 times cheaper than batch 1. Power cost is inversely proportional to batch size — the node uses the same power over the same time while doing 32 times the work.

But power is only part of the cost — add capital

Taking an 8×H100 node at USD 250,000, depreciated over 3 years, at 80% utilization:

$$\frac{250{,}000}{3 \times 8{,}760 \times 0.8} = \$11.9 \text{ /hour}$$

At batch 32, one million tokens takes 0.038 hours, so **capital cost is USD 0.45.**

COST ITEM	PER MILLION TOKENS (BATCH 32)	SHARE
Power	USD 0.023	4.9%
Capital (depreciation)	**USD 0.45**	**95.1%**

Capital cost is about 20 times the power cost.

What this result says — one of the practical conclusions of this book

1. **This is why "don't let the GPUs sit idle" is the first principle.** The dominant cost is a fixed cost per hour, so utilization is cost. The economic value of serving optimizations that enlarge batches and cut waiting time far exceeds that of saving power. The capital cost above is 250,000 / (3 × 8,760) ≈ $9.51 per calendar hour divided by utilization. At 80% utilization it is USD 11.9 per hour used; at 50% it becomes USD 19.0. **Who bears this risk is decided by the contract.** A customer who leases GPU cloud capacity under a take-or-pay commitment pays for the committed capacity whether it is used or not (in the CoreWeave case of section 7, over 98% of 2025 revenue came from such commitments). Then the cost of idle GPUs falls on the customer, not the leasing company, and the company borrows against those commitments to buy the next GPUs. The risk does not disappear; it moves to the customer and, if the commitments are concentrated in a few large customers, to those customers' credit (in the same 10-K, approximately 67% of 2025 revenue came from the single largest customer).
2. **This does not mean power is unimportant.** Power acts not as a cost but as a **constraint** — as Exercise 5 showed, if you cannot bring in electricity, you cannot install accelerators even if you can buy them.
3. **It explains quantitatively why quantization and batch optimization are pursued so aggressively.** Both are ways of getting more tokens out of the same capital.

> **Caution:** the above looks only at power and capital. Real service cost adds labor, network, storage, unused capacity and safety margins, and API prices add margin and recovery of development costs. **This calculation gives a lower bound, not a price.**

The figure below is for batch 32, node power 6.4 kW (assumed) and 80% utilization. In the screen edition you can change the batch, node power (DGX H100 maximum 10.2 kW), utilization (including the 50% of the take-or-pay paragraph) and unit price, and see how the power cost and capital cost per million tokens diverge.



Figure 7-6. Running a 70B INT8 model on an 8×H100 node (26.8 TB/s, bandwidth utilization 0.6) at batch size 32, the power cost of 1 million tokens is about USD 0.023 and the capital cost about USD 0.45, so capital accounts for 95.1%. — In the HTML edition you can change the values here (HTML edition, Session 7, Solution to Exercise 6).

Self-study appendix B — Leads for the discussion questions

QUESTION TYPE	CORE AXIS	LEAD
Is power the real ceiling?	**What can be substituted?**	Semiconductor bottlenecks yield to capital and time. The power grid carries non-market constraints — **siting, permitting and grid stability** — and is different in kind. The lead-time table in Exercise 5(e) is the evidence
Do efficiency gains reduce total demand?	**Jevons paradox**	When energy per unit of computation improves, unit prices fall and demand rises. Historically, cases in IT where efficiency gains reduced total power consumption are rare. But **when growth runs into a physical ceiling,** the story changes
What to watch instead of PUE?	**What goes in the denominator?**	The conclusion of Exercise 3 — use a metric that contains the amount of work (J/token) together with absolute volume (total MW). In regulatory and disclosure contexts, the key point is that **requiring only ratio metrics guarantees they will be gamed**
Siting constraints in Korea	**Grid and cooling water**	Concentration in the Seoul capital region and grid headroom move in opposite directions. The shift to liquid cooling makes water use (WUE) a variable too. Regional dispersion is a heavy constraint for training clusters because of communication latency but is actually an advantage for inference services — **the nature of the workload decides the site.** Tariffs are a variable too. Article 45 of the Special Act on the Promotion of Distributed Energy (in Korean) (headed "Region-specific electricity rates"; current text, in force from 2026-09-18) allows the electricity retailer, for purposes such as balanced national development, to set electricity rates differently when drawing up its basic supply terms, taking transmission and distribution costs and the like into account
Feedback with the memory market	**AI eats into the general market**	The theme of section 7. When HBM production eats into general DRAM capacity, consumer memory prices rise. This **crowding-out between demands** is a structure absent from past cycles, and the strongest argument for discussion question 5 of Session 1

> **Comparable cases elsewhere.** Other jurisdictions have managed new data center load through grid-connection policy or capacity allocation rather than through tariffs. Ireland's energy regulator (CRU) decided on December 12, 2025 that new data centers connecting to the network "will continue to be required to provide generation and/or storage capacity, either onsite or local in proximity to match the requested data centre maximum import demand capacity"; system operators must also consider whether a requested connection "is in a constrained or unconstrained location" (source). Singapore lifted a temporary pause on data center growth in 2022 and, through a pilot call for applications, provisionally awarded about 80 MW to four companies out of more than 20 proposals (EDB, July 14, 2023) (source).

Self-study appendix C — Key terms

TERM	DEFINITION
PUE (power usage effectiveness)	Total facility power ÷ IT power. 1.0 is the ideal
WUE / CUE (water / carbon usage effectiveness)	Water use / carbon emissions per unit of IT energy
White space	The computer-room floor area where racks actually stand
Rack power density	Power drawn by one rack [kW/rack]
Air / liquid cooling	Carrying heat away with air / with liquid
Direct-to-chip liquid cooling	Liquid cooling with a cold plate attached directly to the chip
Immersion cooling	Submerging servers in an insulating fluid
CDU (coolant distribution unit)	A unit that separates and distributes the primary and secondary cooling loops
Cold plate	A metal plate pressed against the chip that passes heat into the coolant
TIM (thermal interface material)	Heat-transfer material between chip and cold plate. A common bottleneck in the chain
Thermal resistance	$R = t\ /\ (kA)$ [K/W]. Adds in series
UPS / PDU / PSU / VRM	Uninterruptible power supply / power distribution unit / power supply unit / voltage regulator module
IR drop	Voltage drop caused by wiring resistance, $\Delta V = IR$
BSPDN (backside power delivery network)	Moving the power wiring to the back of the wafer
PPA (power purchase agreement)	A long-term electricity purchase contract signed directly with a generator
RDMA (remote direct memory access)	A transfer method in which the network card reads and writes another node's memory directly without going through that node's CPU. Used by InfiniBand and RoCE. NVIDIA's GPUDirect RDMA opens a path for PCIe devices such as network cards to exchange data directly with GPU memory
take-or-pay	A contract under which the buyer pays for the committed capacity whether it uses it or not. Utilization risk passes from supplier to buyer
Jevons paradox	The phenomenon in which improved efficiency increases total consumption

Self-study appendix D — Self-assessment

☐ I can use $Q = \dot{m} c_p \Delta T$ to calculate the flow required for any rack power → Exercise 1 ★

☐ I can explain with three reasons **why air cooling is not practical for a 120 kW rack** → Exercise 1 ★

☐ I can explain from material properties why water is 3,500 times better than air by volume → Self-study guide

☐ I can multiply power-chain efficiencies to calculate the power drawn from the grid, and tell **where losses become heat** → Exercise 2

☐ I can show in numbers that **PUE penalizes improvements in IT efficiency** → Exercise 3 ★

☐ I can propose two or more alternatives to PUE and give the reasons → Exercise 3

☐ I can calculate IR drop and explain why BSPDN is needed, linking it to Session 5 → Exercise 4 ★

☐ **I can estimate the power, HBM and rack count of a 10,000-accelerator cluster on my own** → Exercise 5 ★★

☐ I can explain that the real bottleneck of a data center is grid lead time → Exercise 5(e)

☐ I can show by calculation that in inference cost **capital is about 20 times the power cost**, and state what follows → Exercise 6 ★

If you can work the ★★ item (Exercise 5) to the end on your own, you have reached the goal of this book.

Closing — the seven sessions in one line

If you read alone, finish by reconstructing this chain **without looking.** This is the final test.

> 1. A transformer is a mass of matrix multiplications (Session 1).
> 2. But decoding reads the whole set of weights for every token, so performance is set by **bandwidth**, not computation (Session 2).
> 3. Raising bandwidth requires a wide, slow bus, which requires stacking and TSVs, and the yield of that stacking sets the price of HBM (Session 3).
> 4. Meanwhile the chip itself becomes **superlinearly** more expensive with area, so large chips cannot be made (Session 4).
> 5. The physics of lithography (the reticle limit, stochastic defects) tightens that limit further (Session 5).
> 6. So the only way is to **split and join**, and the joining distance in turn sets power (Session 6).
> 7. All that power becomes heat, and the ability to remove the heat and the ability to bring in electricity become the **final ceiling** (Session 7).

If you can attach one number to each arrow from one step to the next (1 → 2, 2 → 3 ...), your study of this book is done.

ARROW	NUMBER TO ATTACH
1 → 2	$I^* \approx 295$ FLOP/byte; at batch 32, utilization 11%
2 → 3	Making 4 TB/s with high-speed pins costs 96 W per stack
3 → 4	With 12-high stacking, 43% of good dies are wasted
4 → 5	8 times the area for 20 times the cost, $A_{50} = \ln 2 \,/\, D_0$
5 → 6	High-NA field 429 mm², 132 ppm per step
6 → 7	200 times the wiring distance = 200 times the energy; interface thermal resistance 20 times that of silicon
7 → end	15.6 MW = 45,600 households; substation lead time 3–7 years

---

Sources

Introductory material chosen by the study curriculum (the "AI data centers" item)

CURRICULUM SUBTOPIC	LINK	HOW THIS BOOK USES IT
AI data centers	IBM Think — "What Is an AI Data Center?"	Introduction to the concept of the AI data center. Basis of section 1. **A general overview suited to opening a presentation** (the Korean edition links IBM's Korean-language version of the page)
AI data centers	Bloom Energy — "Data Center Cooling for Hyperscale and AI Workloads"	Comparison of cooling methods. Direct background to section 3. **Material from a fuel-cell company; read the self-generation parts with its interests in mind**
AI data centers	Bloom Energy — 2026 Data Center Power Report: Mid-Year Pulse	Data center trends in 2026. Macro figures in section 6. **The same caveat applies**
AI data centers and the memory market	Abdullayev, "The Hungry Intelligence: How AI Cannibalized the Global Memory Market and Priced Out the Consumer World (2022–2026)," SSRN Working Paper (2026)	An economics working paper, centered on company cases, on how AI data center demand affected general-purpose DRAM supply and prices. **Direct background to section 7. It has not been peer-reviewed and is mainly qualitative, so recheck its figures against the original sources**
AI data centers and the memory market	Probity Market Insights — HBM market report	HBM market size and outlook. **Market research material, and a target for the "reading numbers critically" exercise in section 4 of Session 1**
AI data centers and the memory market	Counterpoint Research — Global DRAM and HBM Market Share (quarterly)	Tracks quarterly market shares. **Updated regularly, so state the date you consulted it**

CURRICULUM SUBTOPIC	LINK	HOW THIS BOOK USES IT
AI data centers and the memory market	ScienceON report TRKO202500001876	Korea Institute of S&T Evaluation and Planning (KISTEP), "K-Cloud Technology Development Project Using AI Semiconductors," final report of the preliminary feasibility study (2024-08; lead ministry: Ministry of Science and ICT) (in Korean; title translated from Korean). Covers the benefits and economic feasibility of a cloud project built on Korean AI chips. It is not a memory-market analysis

Primary sources added in edition 2.0 (checked 2026-10-08)

- IEA, *Key Questions on Energy and AI* (2026-04) — section 6 global outlook (485 → 950 TWh, around 3%, +17% in 2025), the two-edition comparison in cause 5 (annex table A.4), grid connection waits of five to ten years
- LBNL, "Berkeley Lab Report Evaluates Increase in Electricity Demand from Data Centers" (2025-01-15) — U.S. figures in section 6
- Uptime Institute Global Data Center Survey 2025 (2025-07) — rack density in section 1, PUE 1.54 in section 4
- Google data center PUE page — PUE 1.09 and its measurement boundary in section 4
- ASHRAE TC 9.9, "Emergence and Expansion of Liquid Cooling in Mainstream Data Centers" (2021) — air-cooling limit in section 3 and Exercise 1
- NVIDIA technical blog, "NVIDIA 800 VDC Architecture Will Power the Next Generation of AI Factories" (2025-05-20) — sections 1 and 2, solution to Exercise 4
- NVIDIA DGX B200, DGX B300 and HGX specification pages — Exercise 5 (a) and (c) (the HGX page gives HGX B200 a total of 1.4 TB, i.e., system usable capacity; the source of the 192 GB chip figure is the Blackwell Architecture Technical Brief V2.1). NVIDIA DGX H100/H200 User Guide — 10.2 kW in the solution to Exercise 6
- Broadcom press release, "Tomahawk 6 – Davisson" (2025-10-08) and NVIDIA press release, "Spectrum-X Photonics" (2025-03-18) — CPO in section 5
- CoreWeave FY2025 Form 10-K (company IR copy) — section 7, solution to Exercise 6
- Meta Q4 2025 earnings release (2026-01-28) and Q2 2026 earnings release (2026-07-29) — capital expenditure guidance in section 7
- Korean National Law Information Center, Special Act on the Promotion of Distributed Energy (in Korean) — Article 45 in appendix B
- UF News on the fourth-generation HiPerGator (2025-05) and HiPerGator unveiling (2025-10) — comparison example in Exercise 5
- NVIDIA GPUDirect RDMA documentation — appendix C
- Patterson et al., "TUE, a New Energy-Efficiency Metric Applied at ORNL's Jaguar," ISC 2013 (Springer, pp. 372–382) — TUE and ITUE in the solution to Exercise 3. The relationship between the two metrics (ITUE is the PUE inside the IT equipment; TUE combines the two into the ratio of total energy to the energy used for computation) was confirmed from the abstract. The detailed measurement boundaries in the (subscription) full text were not confirmed

Verified (cross-checked against several sources 2026-08-02; primary sources rechecked and updated 2026-10-08)

- Properties of air and water (density, specific heat, thermal conductivity) and the ratio of volumetric heat capacities — standard property values
- Approximate density ranges by cooling method (air cooling roughly 30 kW, RDHx ~40 kW, D2C tens of kW to 175 kW) — several industry analyses give similar ranges. **The range boundaries differ by source, so they are described as "roughly."** The air-cooling range was corrected on 2026-10-08 from the "~15 kW" of the previous edition. The basis is the ASHRAE TC 9.9 white paper (up to 5,000 CFM for a 40–50 kW rack versus 1,900 CFM per floor tile) and Uptime 2025 (few facilities exceed 30 kW); no single official limit exists in primary sources
- That direct-to-chip cooling (D2C) has become the de facto standard in AI racks — several sources agree
- IEA data center electricity outlook — updated 2026-10-08 from the original of *Key Questions on Energy and AI* (2026-04): 485 TWh in 2025 → 950 TWh in 2030 (Base Case; annex table 945 TWh), around 3% of global electricity in 2030. The 2026 edition also gives its Base Case in the main text (p. 24) as about 415 TWh in 2024 → about 950 TWh in 2030, and annex table A.4 gives 416 TWh for 2024 (checked 2026-10-22). The 2025 edition's path from about 415 TWh in 2024 to about 945 TWh in 2030 is kept as the example in cause 5
- U.S. data center electricity 176 TWh (2023, 4.4%), 325–580 TWh (2028 estimate, 6.7–12%) — original LBNL press release (2026-10-08)
- Measured PUE: Uptime 2025 weighted average 1.54 (n=681, virtually flat for six years), Google 2025 fleet 1.09 (TTM, all overheads included, large facilities in stable operation) — each original (2026-10-08). The same survey's average PUE of 1.48 for facilities commissioned within five years of the survey (from around early 2020) (Uptime 2025 report PDF, p. 8) is used in section 4. Uptime wording quoted in sections 1 and 4 rechecked against the report PDF 2026-10-22
- DGX B200 GPU memory 1,440 GB and maximum about 14.3 kW, DGX B300 2.1 TB and about 14 kW, HGX B300 2.1 TB — NVIDIA specification pages (2026-10-08)
- 800 VDC: 54 V rack distribution, limits above 200 kW, 1 MW racks and 2027, up to 200 kg of copper busbar for a 1 MW rack at 54 V, efficiency up to 5% and TCO up to 30% (company claims) — original NVIDIA technical blog (2026-10-08; quoted wording rechecked 2026-10-22)
- CoreWeave take-or-pay, share of committed-contract revenue (88%, 96%, over 98%), weighted-average contract duration of approximately five years, largest customer approximately 67% — original FY2025 10-K (2026-10-08; quoted wording rechecked 2026-10-22)
- That rack power density is rising steeply — several sources agree. On 2026-10-08, described with dates as two worlds using Uptime 2025 (spread of 10–30 kW racks in ordinary facilities) and NVIDIA (AI racks above 200 kW, a 1 MW target)

Unconfirmed or widely varying (not asserted in the text)

- **Specific figures for total data center power consumption** — vary greatly between institutions. Only the IEA figures for the world and the LBNL figures for the United States are given, with sources (LBNL 2028 as a range), and no other figures are quoted
- **Specific values for average rack power density** — vary greatly by survey body, sample and year. The text carries only Uptime 2025's description of the distribution (spread of 10–30 kW, few above 30 kW) and gives no average
- **Maximum rack power of the latest AI systems** — depends on product and configuration. The text describes only "over 100 kW" and the design direction NVIDIA has stated (above 200 kW, 1 MW in 2027)
- **Share of AI data centers in total data center power** — varies widely because of definitional problems
- **Power, efficiency and accelerator specifications in the section 9 exercises** — all values in the problems are **assumed values for practice**, not the actual values of any particular product or facility. Only the DGX B200, B300 and H100 specifications and the HiPerGator configuration added to the solutions for comparison are figures published by manufacturers or institutions; the flow and temperature difference for water-source cooling are assumed values
- **The crowding-out multiplier of HBM capacity conversion on ordinary DRAM supply** — the mechanism is established, but the quantitative multiplier is each company's trade secret. The text describes only the direction
- **The 3,000 kWh annual electricity consumption per household in Exercise 5 (b)** — a rough value for the calculation exercise. The official national average was not confirmed. Seoul Metropolitan Government energy data (based on KEPCO statistics) give Seoul household electricity consumption of 285 kWh per month in 2023 (about 3,400 kWh a year), so an actual household-count conversion could come out smaller than "45,600 households" (checked 2026-10-22)
- **The lead-time table in Exercise 5 (e) (months to 1 year, 1–2 years, 3–7 years)** — rough orders of magnitude. The only primary figures are the IEA's (2026-04) transformer lead times of two to three years and grid-connection waits of five to ten years

APPENDIX

Appendix I — Formula Summary and Exercise Index

This appendix gathers, session by session, the equations the book actually uses, and attaches to each the conditions under which it holds. **If you can rebuild these tables without looking, the book has done its job.**

1. Compute and bandwidth (Session 2)

EQUATION	MEANING	WATCH OUT FOR
$P_{\text{attainable}}(I) = \min(P_{\text{peak}}, B \times I)$	**Roofline.** The performance ceiling is the lower of the compute limit and the bandwidth limit	It bounds a kernel in steady state. It says nothing about latency, communication or overhead
$I^* = P_{\text{peak}} / B$	**Ridge point.** Where compute-bound and memory-bound split	Always compute it from dense specifications
$I \approx M$ (prefill)	Weights read once serve M tokens. I is in FLOP/byte	An approximation for when reading weights dominates memory traffic and each weight takes 2 bytes (fp16). Activation reads and writes are left out (activations kept on chip or fused). In general $I \approx 2M / b$. Because M is large, prefill is compute-bound
$I \approx B_{\text{sz}}$ (decode)	Weights read once serve only as many tokens as the batch size	Same conditions as above (2-byte weights, weight reads dominate). In general $I \approx 2B_{\text{sz}} / b$, and the boundary where decode becomes compute-bound is $B_{\text{sz}} = (b / 2)\, I^*$ (compute taken at BF16 peak). Small batches are memory-bound
$\text{tokens/s} \lesssim \frac{B}{P \cdot b} \times B_{\text{sz}}$	**Decode ceiling.** P is the parameter count, b bytes per parameter	B is the combined bandwidth of the N chips the model is split across. In practice, utilization of 60–80% applies and communication overhead comes on top
$\text{KV} = 4 \cdot L \cdot S \cdot d \cdot B_{\text{sz}}$ bytes	Size of an fp16 KV cache	d is the total dimension of the KV heads. With GQA it shrinks a great deal

Numbers to remember: for current accelerators, I^* is **around 300 FLOP/byte.** At a decode batch of 32, utilization is about 11%.

2. Yield and cost (Session 4)

EQUATION	MEANING
$Y = e^{-AD_0}$	**Poisson yield.** A is die area [cm²], D_0 defect density [defects/cm²]
$Y = \left(1 + \frac{AD_0}{\alpha}\right)^{-\alpha}$	**Negative binomial yield.** Accounts for defect clustering. As $\alpha \to \infty$ it becomes Poisson
$\text{GDPW} \approx \frac{\pi d^2 / 4}{A} - \frac{\pi d}{\sqrt{2A}}$	Dies per wafer. The second term is the edge loss. d is the wafer diameter, and A must be in the square of the same length unit as d (A in cm² if $d = 30$ cm, in mm² if $d = 300$ mm)
$\text{Cost per die} = \frac{W}{\text{GDPW} \times Y}$	W = price of one wafer
$A_{50} = \frac{\ln 2}{D_0}$	Die area at which yield falls to 50%
$N^* = \frac{C_{\text{NRE}}}{(c_G + o_G) - (c_A + o_A)}$	ASIC break-even volume. Assumes one ASIC does the work of one GPU at the same throughput and quality (one-for-one replacement)
$Y_{\text{stack}} = y^n \times \prod y_{\text{bond}}$	**Stack yield** (Session 3)

Numbers to remember: at $D_0 = 0.1$, an 800 mm² die yields **45%**. A die with 8 times the area costs **20 times** as much. If D_0 doubles, the die area allowed for the same target yield halves.

Why clustering raises yield: e^{-x} is convex, so by Jensen's inequality $E[e^{-\Lambda}] \geq e^{-E[\Lambda]}$. With the same total number of defects, when they bunch together the remaining dies are clean.

---

3. Lithography (Session 5)

EQUATION	MEANING
$\text{CD} = k_1 \frac{\lambda}{\text{NA}}$	**Rayleigh criterion.** The smallest line width that can be resolved
$\text{DOF} = k_2 \frac{\lambda}{\text{NA}^2}$	Depth of focus. Inversely proportional to **the square of NA**
$E = hc / \lambda$	Photon energy. At the same dose (energy per unit area), a shorter wavelength means fewer photons
$\text{Relative fluctuation} = 1 / \sqrt{N}$	The root of stochastic defects
$y_{\text{step}} = Y^{1/n}$	Step yield. n is the number of process steps

Numbers to remember: at $k_1 = 0.3$, ArF immersion gives **42.9 nm**, EUV **12.3 nm**, High-NA **7.4 nm**. High-NA DOF is **0.36 times** that of EUV. EUV delivers **7%** as many photons as ArF, with **3.78 times** the fluctuation. To hold line yield at 90% over 800 steps, each step may fail at no more than **131.7 ppm**.

---

4. Packaging and heat (Session 6)

EQUATION	MEANING
$E = \frac{1}{2}CV^2$	Energy to send a signal over a wire. C is roughly proportional to wire length
$R = \frac{t}{kA}$	Thermal resistance. t thickness, k thermal conductivity, A area
$\Delta T = Q \times \sum R$	When the same heat flow Q [W] passes through every part of a series path, thermal resistances **add in series.** If each layer generates its own heat so that the flow differs from segment to segment, $\Delta T = \sum Q_i R_i$
Required area $= N_{\text{wires}} \times (\text{pitch})^2$	Area needed for an area-array interconnect

Numbers to remember: the energy ratio between a substrate trace (10 mm) and a TSV (50 µm) is **200 times**. In a 4-layer stack, the accumulated temperature difference ΔT across the silicon layers is **3.3 K** against **66.7 K** across the bonding interfaces: the interfaces are the bottleneck. Silicon conducts heat at about **150 W/(m·K)**; underfill and bump layers at **1–5**.

---

5. Data centers (Session 7)

EQUATION	MEANING
$Q = \dot{m}\, c_p\, \Delta T$	Rate of heat carried by a fluid (W)
$\text{PUE} = \frac{\text{total facility power}}{\text{IT equipment power}}$	1.0 is the ideal
$\Delta V = IR$, loss $= I^2R$	IR drop and power-network loss
Overall efficiency $= \prod \eta_i$	Efficiency erodes multiplicatively along the power chain

Numbers to remember: specific heat of air **1,005**, of water **4,186** J/(kg·K). **Per unit volume, water holds about 3,500 times as much heat as air.** A 120 kW rack: air cooling needs 6.75 m^3/s (an air speed of about 5.6 m/s across the rack face, outside the practical range) against **172 L/min** for liquid cooling. Overall efficiency of the power chain is about **80%**. 10,000 accelerators = **15.6 MW** = 137 GWh a year.

---

6. Index of quantitative exercises (all 32)

SESSION	EXERCISE	WHAT IT COMPUTES	DIFFICULTY
1	1	The definitional difference that splits a market share into 90% and 70%	Easy
1	2	Why 200% growth and 25% growth do not contradict each other	Easy
1	3	Inferring where the bottleneck is from three observations	Medium
2	**1**	**Ridge points across three generations**	Easy ★
2	**2**	**The decoding ceiling: model × precision × number of chips**	Medium ★
2	3	KV cache budget and maximum batch	Medium
2	4	ASIC break-even volume	Medium
3	**1**	**Stack yield and the good dies thrown away**	Medium ★
3	2	Bandwidth design and pJ/bit power	Medium
3	3	Optimizing how a memory budget is split	Hard
3	4	Criteria for judging whether HBF fits	Qualitative
4	1	The yield curve and the area where yield collapses to 50%	Easy
4	**2**	**Cost per die: 8 times the area, 20 times the cost**	Medium ★
4	3	Poisson vs. negative binomial (Jensen's inequality)	Hard
4	4	Chiplet profit and loss, and the crossover point	Medium
4	5	The ASIC decision and a realistic team size	Easy
5	1	The Rayleigh criterion across three generations	Easy
5	2	The price paid in depth of focus	Medium
5	3	Stochastic defects: photon count and dose	Hard
5	**4**	**Step yield: how unforgiving 800 steps are**	Easy ★
5	5	Exposure field and maximum die size	Medium
6	1	Searching for the best number of chiplets	Hard
6	2	Wiring energy, 200 times	Easy
6	3	Interposer area and the reticle limit	Medium
6	4	Heat in a 3D stack: silicon against interfaces	Medium
6	5	Number of connections and the pitch they require	Medium
7	1	Coolant flow: air against liquid	Medium
7	2	Power-chain losses and where the heat load lands	Easy
7	3	The structural trap in PUE	Medium
7	4	IR drop and BSPDN	Easy

SESSION	EXERCISE	WHAT IT COMPUTES	DIFFICULTY
7	**5**	**Estimating the scale of a data center (synthesis)**	Hard ★★
7	6	Inference cost: capital against electricity	Medium

The five starred exercises are the backbone of the book. **If you can solve the double-starred one (7-5) on your own from start to finish, you have reached the goal.**

---

7. The seven sessions in one line

> A transformer is a mass of matrix multiplications (Session 1) → but decoding reads every weight for each token, so performance is set by **bandwidth**, not compute (Session 2) → raising bandwidth takes a wide, slow bus, which takes **stacking and TSVs**, and the yield of that stacking sets the price of HBM (Session 3) → meanwhile a chip gets **superlinearly** more expensive with area (Session 4), so you cannot build a big one → and the physics of lithography tightens that limit further (Session 5) → so the industry leans toward **splitting chips and joining them**, and the distance over which they are joined sets power once again (Session 6) → all that power ends up as heat, and the ability to remove heat and the ability to bring in electricity become the **final ceiling** (Session 7).

ARROW	NUMBER TO ATTACH
1 → 2	$I^* \approx 295$ FLOP/byte; at a batch of 32, utilization 11%
2 → 3	Building 4 TB/s from high-speed pins costs 96 W per stack
3 → 4	In a 12-high stack, 43% of good dies are wasted
4 → 5	8 times the area, 20 times the cost; $A_{50} = \ln 2 \,/\, D_0$
5 → 6	High-NA field 429 mm², 132 ppm per step
6 → 7	200 times the wiring distance = 200 times the energy; interface thermal resistance 20 times that of silicon
7 → end	15.6 MW = 45,600 households; substation lead time 3–7 years

APPENDIX

Appendix II — Glossary Index

This appendix gathers the terms that appear across the seven sessions, in alphabetical order. The **Session** column shows where each term is first defined; if the definition here is too brief, go to that session's self-study appendix C.

TERM	DEFINITION	SESSION
2.5D	Placing dies side by side on an interposer	6
3D stacking	Stacking dies vertically	6
accelerator	Umbrella term for chips specialized for particular computations (GPU, TPU, NPU and others)	1
air / liquid cooling	Carrying heat away with air / with liquid	7
ALD (atomic layer deposition)	Deposition one atomic layer at a time for precise thickness control	5
anisotropy / selectivity	How strictly etching goes only downward / how strictly it removes only the intended material	5
arithmetic intensity	Operations performed per byte moved from memory [FLOP/byte]	1
ASIC (application-specific integrated circuit)	A custom chip built for a fixed purpose	1
back-end (assembly & test), packaging	The stage that cuts the wafer, turns chips into packages and tests them. Not the same as BEOL, the forming of wiring layers inside the wafer process	6
bandwidth / capacity	How much is moved per second / how much is held. **The two axes to keep apart throughout Session 3**	3
bank / channel	Units of parallel access inside DRAM	3
batch size	The number of sequences processed at once	2
binning	Testing dies and sorting them into grades. Defect binning switches off some faulty cores and sells the die as a lower grade (yield recovery); speed and power binning sorts defect-free dies by maximum operating frequency and power	4
BSPDN (backside power delivery network)	Moving the power wiring to the back side of the wafer	5, 7
bullwhip effect	Demand swings that grow larger the further upstream they travel in a supply chain	1
CD (critical dimension)	The smallest line width that must be made	5
CDU (coolant distribution unit)	A unit that separates the primary and secondary cooling loops and distributes coolant	7
CFET (complementary FET)	A next-generation structure that stacks n-type and p-type transistors vertically	5

TERM	DEFINITION	SESSION
chiplet	Splitting a large chip into several small dies, making them separately and joining them	4
clustering parameter (α)	The smaller it is, the more defects bunch together. At ∞ the model becomes Poisson	4
CMP (chemical mechanical polishing)	Flattening a surface by chemical and mechanical means	5
cold plate	A metal plate pressed against a chip that passes its heat to coolant	7
compute-bound	A state in which the compute units are the limit ($I > I^*$)	2
CoWoS (chip-on-wafer-on-substrate)	TSMC's flagship 2.5D advanced-packaging method	1, 6
defect density (D_0)	Number of killer defects per unit area [defects/cm^2]	4
deposition	A process that lays down a needed film	5
depth of focus (DOF)	The range of depth over which focus holds. $\propto \lambda$ / NA^2	5
dicing	Cutting a wafer into individual dies	6
die	A single chip cut from a wafer	4
direct-to-chip liquid cooling	Liquid cooling with cold plates attached directly to the chip	7
dual damascene	A wiring process that etches vias and trenches into the dielectric, fills them with copper in one step, and removes the excess by CMP	5
EDA (electronic design automation)	Software tools for chip design	1
EMIB / bridge (embedded bridge)	Connecting only the regions that need it with a small piece of silicon	6
etching	A process that removes unwanted material	5
EUV (extreme ultraviolet)	Lithography with extreme ultraviolet light at a wavelength of 13.5 nm	5
fabless	A semiconductor company that designs chips and outsources their manufacture	1
FlashAttention	A representative kernel that fuses attention to cut round trips to HBM	2
flip chip	Turning a die over and attaching it to the substrate directly through bumps	6
foundry	A plant that manufactures other companies' designs under contract	1
front-end process (FEOL + BEOL)	The whole wafer process that builds transistors (FEOL) and the wiring layers above them (BEOL) on the wafer. Distinct from the back end, which cuts and joins wafers	5
GAA (gate-all-around)	A transistor structure in which the gate wraps the channel on all sides	5
GDPW (gross die per wafer)	Total number of dies from one wafer (before yield is applied)	4
GEMM (general matrix multiply)	General matrix–matrix multiplication	2
good die	A die that works. GDPW × yield	4

TERM	DEFINITION	SESSION
GQA / MQA (grouped / multi-query attention)	Attention variants that reduce the number of KV heads to shrink the KV cache	2
HAR etch (high aspect ratio etch)	Etching deep, narrow holes. A hard problem in 3D NAND and TSVs	5
HBF (high bandwidth flash)	A proposal to stack NAND the same way to reach large capacity	3
HBM (high bandwidth memory)	Memory that stacks DRAM vertically to maximize bandwidth	1, 3
heterogeneous integration	Combining dies of different processes and functions in one package	6
High-k	A gate insulator with a high dielectric constant. It blocks leakage current	5
High-NA EUV	Next-generation EUV with NA raised from 0.33 to 0.55. The field is half the size	5
hybrid bonding	Bonding copper directly to copper without bumps. Better on both pitch and thermal resistance	3, 6
hyperscaler	A very large cloud operator (Google, AWS, Microsoft, Meta and others)	1
IDM (integrated device manufacturer)	A semiconductor company that designs, manufactures and sells	1, 4
immersion cooling	Submerging servers in an insulating fluid	7
inspection / metrology	Finding whether there are defects / measuring dimensions and thickness	5
interposer	An intermediate substrate that carries several dies and connects them	3, 6
ion implantation	Driving dopant ions into the material to change its electrical properties	5
IR drop	Voltage drop caused by wiring resistance, $\Delta V = IR$	7
Jevons paradox	Better efficiency leading to higher total consumption	7
kernel fusion	An optimization that merges several operations so intermediate results are not written to memory	2
KGD (known good die)	A die verified as good before stacking	3, 6
KV cache (key-value cache)	Memory that stores intermediate results for tokens already produced, to avoid recomputing them	1
memory-bound	A state in which memory transfer is the limit ($I < I^*$)	2
memory tiering	Placing data on different media according to how often it is accessed	3
microbump	A tiny solder bump that joins one die to another	3, 6
MoE (mixture of experts)	A sparse model structure that activates only some experts for each token	2
multi-patterning	Exposing one layer in several passes to get around the resolution limit	5
negative binomial yield model	$Y = (1 + AD_0 \,/\, \alpha)^{-\alpha}$. Accounts for clustering	4
node (process node)	The name of a process generation, such as "3nm". A marketing label, not a physical dimension	1
NRE (non-recurring engineering)	The one-time cost of developing a chip	2, 4

TERM	DEFINITION	SESSION
numerical aperture (NA)	A measure of the angle over which a lens gathers light	5
OPC (optical proximity correction)	Altering the mask in advance to cancel diffraction distortion	5
OSAT (outsourced semiconductor assembly and test)	A contractor specializing in the back end (packaging and test)	1
PDK (process design kit)	The bundle of process-specific design rules and models a foundry supplies	4
photoresist	A light-sensitive film whose properties change when exposed	5
PIM / PNM (processing in / near memory)	Structures that put compute units inside memory / next to memory	3
pJ/bit (picojoule per bit)	Energy to move 1 bit. The practical constraint in bandwidth design	3
Poisson yield model	$Y = e^{-AD_0}$. Assumes defects are completely random	4
PPA (power purchase agreement)	A long-term electricity purchase contract signed directly with a generator	7
prefill / decode	The stage that processes the whole input at once / the stage that generates tokens one at a time	2
PUE (power usage effectiveness)	Total facility power ÷ IT power. 1.0 is the ideal	7
quantization	Representing weights with fewer bits to save capacity and bandwidth	1
rack power density	Power drawn by one rack [kW/rack]	7
Rayleigh criterion	$\mathrm{CD} = k_1\lambda$ / NA. The fundamental limit on resolution	5
RDL (redistribution layer)	A wiring layer that relocates pads	6
reduction projection	Exposing a mask pattern onto the wafer at reduced size. 4× is standard; High-NA uses 4× and 8×	5
refresh	Rewriting the charge in DRAM capacitors before it leaks away	3
reticle limit	The largest area a lithography tool exposes in one shot (about 26×33 mm)	4
ridge point	The arithmetic intensity at which compute-bound and memory-bound split, $I^* = P_{\text{peak}} / B$	2
roofline model	A model that expresses the performance ceiling as the minimum of the compute limit and the bandwidth limit	2
signal integrity	The degree to which a signal arrives free of distortion and noise	6
single point of failure	A point whose failure alone stops the whole system	1
stack yield	The probability that a whole stack works. Falls steeply as y^n	3
stochastic defect	A random defect caused by too few photons	5
structured ASIC	An intermediate form that customizes only some layers to lower NRE	4
structured sparsity	Setting some weights to 0 to cut computation. The main culprit behind inflated spec-sheet figures	2

TERM	DEFINITION	SESSION
systolic array	A compute structure that streams data through a grid, repeating multiply-accumulate	2
tensor parallelism	Splitting the matrices of one layer across several accelerators	2
TGV (through-glass via)	A via through a glass substrate. A candidate to replace silicon	6
thermal oxidation	Growing an SiO_2 film by oxidizing silicon at high temperature. The thickness follows the Deal–Grove equation	5
thermal resistance	$R = t \,/\, (kA)$ [K/W]. Adds in series	7
tiling / blocking	An optimization that cuts data into pieces small enough to sit on chip and reuses them	2
TIM (thermal interface material)	The material that carries heat between a chip and its heat sink	6, 7
training / inference	Building a model / using a model that has been built	1
TSV (through-silicon via)	A vertical connection through silicon	3, 6
UCIe (universal chiplet interconnect express)	An interface standard between chiplets	6
underfill	A resin that fills the space between bumps for mechanical reinforcement	6
UPS / PDU / PSU / VRM	Uninterruptible power supply / power distribution unit / power supply unit / voltage regulator module	7
warpage	Bending of a package caused by differences in thermal expansion	6
white space	The floor area of the computer room where racks actually stand	7
WUE / CUE (water / carbon usage effectiveness)	Water used / carbon emitted per unit of IT energy	7
yield	The fraction of what is made that works	1
yield learning	The process by which D_0 falls over time	4

APPENDIX

Appendix III — List of Interactive Figures

At the places listed below, the HTML edition lets you change a figure's values yourself. The print edition carries a still image of the starting state in the same place. Each standalone page adds an explanation and a check table that matches the figure's calculations against the values in the text and solutions.

FIGURE	LOCATION	CONTENT	STANDALONE PAGE
Figure 2-1	Session 2 Section 4	The fourth beat of one token (M = 1, batch-1 decode) flowing through a 4×4 weight-stationary systolic array. While the activations pass along the diagonal, only the 4 PEs on the diagonal out of 16 are working.	widgets/systolic.html
Figure 2-2	Session 2 Section 6	Decode workload on the roofline. On an H100 at batch size 32 with fp16 weights, arithmetic intensity is about 32, short of the ridge point (about 295), so the workload is memory-bound and the compute-unit utilization ceiling is about 11%.	widgets/roofline.html
Figure 2-3	Session 2 Solution to Exercise 3	Load a 70B-class model onto one H100 80GB with INT8 weights (70 GB), take sequences of S = 4,096 with GQA and fp16 KV, and the KV cache has room for only 7 sequences. At batch size 7, assuming the math also runs in INT8, compute-unit utilization is about 2.4%.	widgets/kv-cache.html
Figure 2-4	Session 2 Solution to Exercise 4	With the chip boundary (GPU 700 W, ASIC 300 W) and the assumptions of Exercise 4 (NRE USD 100 million, unit prices USD 3,000 and 30,000, PUE 1.3, 80% utilization, USD 0.08/kWh, 3 years), the two total-cost lines meet at N* ≈ 3,590 units. Of the USD 27,875 denominator, the power-cost difference is USD 875 (3%).	widgets/asic-breakeven.html
Figure 3-1	Session 3 Section 6	If 10% of the bytes read come from HBF (0.4 TB/s, assumed), the effective bandwidth mixed with an HBM4 stack (3.3 TB/s) is about 1.91 TB/s, 58% of reading from HBM alone, and capacity is 548 GB. The band is the 0.4–3.0 TB/s range of published HBF specs.	widgets/hbm-hbf-tier.html
Figure 3-2	Session 3 Solution to Exercise 1	Stacking 12 dies with a die yield of 0.95 and no screening gives a stack yield of 54.0%, and each failed stack throws away 10.7 good dies on average. The bars are 100 randomly drawn stacks.	widgets/hbm.html

FIGURE	LOCATION	CONTENT	STANDALONE PAGE
Figure 3-3	Session 3 Solution to Exercise 2	Delivering 4 TB/s per stack over a 1,024-bit bus takes 31.25 Gbps per pin; over a 2,048-bit bus, 15.6 Gbps. With energy per bit set to 3 and 0.75 pJ/bit, memory-interface power for 8 stacks approaches 800 W for (a) and is about 190 W for (b) (about 130–260 W over the 0.5–1 pJ/bit range).	widgets/hbm.html
Figure 4-1	Session 4 Solution to Exercise 2	Laying 800 mm^2 dies on a 300 mm wafer gives a direct count of 68 (66–71 depending on grid position); the $\sqrt{(2A)}$ approximation gives 64.8. With $D_0 = 0.1$, yield is 44.9%, good dies number 29.1, and cost per die is 0.03435 times the wafer price, about 19.9 times that of a 100 mm^2 die. The defect dots are one random sample.	widgets/die-cost.html
Figure 4-2	Session 4 Solution to Exercise 3	At $\lambda = 0.8$, splitting the die into 10 cells gives $(1 - \lambda/n)^n = 0.4344$, 97% of the limit $\exp(-0.8) = 0.4493$. At the same λ, the negative binomial yield ($\alpha = 2$) is 51.0%, higher than the Poisson 44.9%, and the variance of defect density is $D_0^2/\alpha = 0.005$.	widgets/die-cost.html
Figure 4-3	Session 4 Solution to Exercise 5	With the assumptions of Exercise 5 (NRE USD 150 million, ASIC USD 4,000 and GPU USD 30,000, 3-year power cost USD 2,000 and 6,000), $N^* = 5{,}000$ units, which falls inside the band "in-house AI infrastructure at a large company: hundreds to thousands of units."	widgets/asic-breakeven.html
Figure 5-1	Session 5 Solution to Exercise 2	At $k_1 = 0.3$, CD is 42.9 nm for ArF immersion, 12.3 nm for EUV and 7.4 nm for High-NA EUV. Depth of focus with $k_2 = 0.5$ is 53, 62 and 22 nm, so High-NA gets 0.36 times the EUV value.	widgets/litho.html
Figure 5-2	Session 5 Solution to Exercise 3	At the same dose, EUV delivers 7% as many photons as ArF, and the relative fluctuation is 3.78 times larger. Raising the dose 14.3 times to match the fluctuation makes exposure 14.3 times longer at the same source power; holding exposure time instead requires source power to rise from 250 W to about 3.6 kW.	widgets/litho.html
Figure 5-3	Session 5 Solution to Exercise 4	To reach 90% line yield over 800 steps, the allowed defect rate per step is 131.7 ppm; over 1,200 steps, 87.8 ppm.	widgets/litho.html
Figure 5-4	Session 5 Solution to Exercise 5	Shrinking the 104 × 132 mm pattern area on the mask 4× in each direction gives a 26 × 33 mm wafer field (858 mm^2); High-NA, which shrinks one direction 8×, gives 26 × 16.5 mm (429 mm^2). With $D_0 = 0.1$, the maximum die yields are 42.4% and 65.1%.	widgets/litho.html
Figure 6-1	Session 6 Section 6	Stacking 8 layers with die yield 0.9 and bonding yield 0.995 (assumed) gives a W2W yield of 0.416, while D2W, which bonds only good dies, gives 0.966: a 2.32-fold difference.	widgets/chiplet.html

FIGURE	LOCATION	CONTENT	STANDALONE PAGE
Figure 6-2	Session 6 Section 7	Removing 1,000 W from a chip to 35 °C ambient through a chain of θJC 0.015, θTIM 0.025 and θSA 0.020 K/W (assumed) gives stage temperature differences of 15, 25 and 20 K and a junction temperature of 95 °C.	widgets/thermal.html
Figure 6-3	Session 6 Section 7	At an activation energy of 0.7 eV, a test at 125 °C and use at 55 °C, the acceleration factor is about 77.7, and 1,000 hours of HTOL equals about 8.87 years. At 0.5 eV it is about 2.56 years; at 0.9 eV, about 30.7 years.	widgets/reliability.html
Figure 6-4	Session 6 Solution to Exercise 1	System cost (as a multiple of the wafer price) when 800 mm^2 of silicon is split into n dies. With D_0 = 0.1, 5% overhead per split and assembly yield 0.99 for each added die, n = 4 is cheapest at 0.01970, 42.6% cheaper than a single die.	widgets/chiplet.html
Figure 6-5	Session 6 Solution to Exercise 2	A 10 mm wire and a 50 µm TSV differ in length by 200 times, so energy per bit also differs by 200 times. At 0.05 pJ/bit for the TSV, 10^{12} bits per second costs 0.05 W versus 10 W.	widgets/thermal.html
Figure 6-6	Session 6 Solution to Exercise 4	Stacking 4 layers of 1 W/mm^2 chips produces a total temperature difference of 3.33 K across the silicon (50 µm) but 66.7 K across interfaces such as bumps and underfill (20 µm, 3 W/(m·K)), 20 times as much.	widgets/thermal.html
Figure 7-1	Session 7 Solution to Exercise 1	Cooling a 120 kW rack with air (ΔT 15 K) takes 6.75 m^3/s (about 14,300 CFM) and an air velocity of 5.6 m/s across the rack face. Liquid cooling (ΔT 10 K) needs only 172 L/min. The two dashed lines are the 5,000 and 1,900 CFM figures from the ASHRAE white paper.	widgets/datacenter.html
Figure 7-2	Session 7 Solution to Exercise 2	With transformer, UPS, PDU, PSU and VRM efficiencies of 99, 96, 99, 94 and 90%, overall efficiency is 0.796, and delivering 1 MW to the chips takes 1.256 MW from the grid. Of the 256 kW lost, the VRM share of 111 kW is the largest.	widgets/datacenter.html
Figure 7-3	Session 7 Solution to Exercise 3	Cutting IT load from 10 MW to 8 MW for the same work lowers total power 14 → 11.44 MW, 18% less, but the fixed load of the facility equipment pushes PUE the wrong way, from 1.40 to 1.43.	widgets/datacenter.html
Figure 7-4	Session 7 Solution to Exercise 4	Feeding a 1 MW rack at 54 V takes about 18.5 kA; at 800 V, 1,250 A. The current is 14.8 times higher, so resistive loss in the same conductor is about 219 times higher.	widgets/datacenter.html

FIGURE	LOCATION	CONTENT	STANDALONE PAGE
Figure 7-5	Session 7 Solution to Exercise 5	10,000 accelerators (1 kW each plus 30% overhead) mean 13 MW of IT load, 15.6 MW for the facility at PUE 1.2, and about 137 GWh a year. Counted with the DGX B200 and B300 system maximums, IT is 17.875 and 17.5 MW, and at 192 GB per chip HBM totals 1.92 PB.	widgets/datacenter.html
Figure 7-6	Session 7 Solution to Exercise 6	Running a 70B INT8 model on an 8×H100 node (26.8 TB/s, bandwidth utilization 0.6) at batch size 32, the power cost of 1 million tokens is about USD 0.023 and the capital cost about USD 0.45, so capital accounts for 95.1%.	widgets/datacenter.html

COLOPHON

Credits

ITEM	DETAILS
Study group	*Sunday AI* · semiconductor sessions
Title	AI Infrastructure and Semiconductors
Editor	Bowon Kwon · Presiding Judge, Daegu District Court
Edition	Edition 2.0.1-en · translated from Korean edition 2.0.1 (2026-10-22)
License	To the extent the editor holds rights in it, this book is released under CC BY-NC 4.0 (Creative Commons Attribution-NonCommercial 4.0 International). A summary is in the first box of "How to Use This Book"; the license text governs. The code of the interactive figures in the screen edition (HTML, CSS and JavaScript) is under the PolyForm Noncommercial License 1.0.0; commercial use is by separate agreement. Tables, quotations and figures reproduced or redrawn from original works are not covered by these licenses and remain subject to the rights of their authors. If a rights holder tells us, we will correct or remove the material. © 2026 Bowon Kwon
Contents	How to Use This Book · Sessions 1–7 · three appendices (formula summary and exercise index, glossary, list of interactive figures) · colophon
Size	32 quantitative exercises · 12 interactive figures · 114 terms

About this book

This self-study guide was made for the semiconductor sessions of the study group **Sunday AI**.

It began as study material, but it was reworked so that **a reader working alone, outside any group, can get to the end.** Each session gives a reading order and a time budget, every quantitative exercise comes with a full solution, and each session closes with a checklist for testing yourself.

How this book was made

The editor made this book with Claude Code. Because it is text produced with generative AI, errors may remain even after it was checked against primary sources.

- Material from the study group's presentations and discussions went in only after it was confirmed against primary sources such as manufacturer announcements, standards documents, agency reports and papers, and was then written afresh.
- Each session was checked independently. A process separate from the one that revised the text received only the revised text and the primary sources, compared the changed sentences, and fixed what was wrong. The numbers in the calculation exercises were recalculated.

- Finally, an AI model from a different company (OpenAI Codex), with no knowledge of how the work had been done, read the whole manuscript, the print edition and the figure code. Each of its 125 findings was checked against the originals and by recalculation, and was either fixed or answered with a written reason for leaving it.
- The interactive figures move only the variables of equations and tables that appear in the text. Each figure's results are checked automatically against the worked solutions in the text, and the book will not build if any of them disagree. The static snapshots in the print edition are the screen-edition figures opened in their initial state and captured.
- Equations were converted to MathML. No external scripts or fonts are loaded.
- Diagrams are drawn in inline SVG. They look the same on screen and in print, in light and dark mode.
- Page numbers, running heads and table-of-contents page numbers in the print edition were computed at typesetting.

How this English edition was made

This English edition was translated from Korean edition 2.0.1 (October 22, 2026) by the editor working with Claude Code. Quotations that were originally in English were restored from the original sources rather than translated back from Korean. Korean-language sources are kept with translated titles and marked "(in Korean)." Numbers were carried over unchanged and checked against the Korean edition; only their notation changed where English needs it (for example, Korean counting units written as thousands or billions, and amounts written as USD). In a few places, boxes headed "Comparable cases elsewhere" add examples from other countries that are not in the Korean edition; each was checked against the primary source cited in the box. Before publication, an AI model from a different company (OpenAI Codex), given only the Korean and English texts and the sources they cite, compared the two editions sentence by sentence; each of its findings was rechecked against the originals and either fixed or answered with a written reason for leaving it.

Edition history

EDITION	DATE	SUMMARY
Study notes	2026-08-02	First draft of the seven sessions for the semiconductor sessions of *Sunday AI* (learning objectives, main text and quantitative exercises for each session)
Study notes, expanded	2026-08-23	For readers working alone, added to each session a self-study guide, solutions to the exercises (32 problems), pointers for the discussion questions, key terms and a self-assessment
Edition 1	2026-08-23 (cover date) to 2026-08-24 (final build)	**Internal edition for the study group (not published).** Bound the seven sessions into one HTML and PDF volume (149 pages), wrote the guide to using the book, the formula summary and a glossary of 111 terms, and redrew 7 diagrams

EDITION	DATE	SUMMARY
Edition 2.0	2026-10-08	**First public edition.** Building on edition 1, added material from the study group's presentations and discussions after confirming it against primary sources, and added 12 interactive figures (screen edition; static snapshots at the same places in the PDF) and links to primary sources. Rechecked numbers that had gone out of date, and corrected calculation and wording errors and overstatements found when another AI model read the whole book. The glossary has 114 terms
Edition 2.0, corrected	2026-10-13	Revised the license wording (the scope of the editor's rights, the bar on adding restrictions through legal terms or technological measures, the status of the summary, and the exclusion of third-party material). The code of the interactive figures moved from CC BY-NC 4.0 to the PolyForm Noncommercial License 1.0.0, with commercial use by separate agreement. Copies received before this date remain under the earlier terms. The content of the book did not change
Edition 2.0.1	2026-10-22	Corrected the volume-production timing of TSMC's A16 process. Where the book had given only the "production in 2026" plan from the 2024 announcement, it now gives "volume production expected to start in 2027," based on a TSMC executive's remarks reported from the April 2026 Technology Symposium (Session 4, the table in section 4; Session 5, section 3; and their sources and verification status). Factual and bibliographic errors found while translating the English edition and checking it against the sources were corrected at the same time, and company names were brought into line with each company's current styling (Sandisk, Arm, ZEISS)
Edition 2.0.1-en	2026-10-22	**English edition.** Translated from the Korean edition 2.0.1 (2026-10-22), which includes the correction of 2026-10-13. Content is the same as the Korean edition; "Comparable cases elsewhere" boxes were added in a few places. Compared against the Korean edition by OpenAI Codex before publication

Acknowledgments

Since January 2022 the editor has been a member of *Sunday AI*, a study group of graduates and students of the Department of Statistics and Data Science at Korea National Open University. Its members propose a new topic each time one ends, and despite busy lives they take turns presenting and share what they know every week. The editor owes them a great deal and is always grateful. What came out of the presentations and discussions was checked again against primary sources before it went in, and any errors that remain are the editor's.

When you use this book

- **Do not quote the numbers as they stand.** Prices, electricity costs and similar inputs in the calculation exercises are assumed values meant to show the direction of a conclusion, and accelerator specifications come from public spec sheets that change with each generation. The details are in "A note on the numbers" at the front of the book.
- **When the numbers date from.** Volume-production and shipment status, power-demand projections, standards and product specifications were rechecked as of October 8, 2026. Announcements made after that date are not reflected.
- **Links.** The links in the text were checked on October 8, 2026. Company blogs and press releases may disappear or move.
- **Errors and revisions.** Please look especially at the arithmetic and units in the calculation exercises and at the specification figures quoted. If you find a statement that differs from its source, please report it through the issue tracker of the public repository. Corrections

will be made in a new edition with a new date.

- **A personal study resource.** The editor compiled this book as part of personal study. It is unrelated to the views or work of any institution the editor belongs to.

Sunday AI · semiconductor sessions · edited by Bowon Kwon · edition 2.0.1-en · translated from Korean edition 2.0.1 (October 22, 2026)